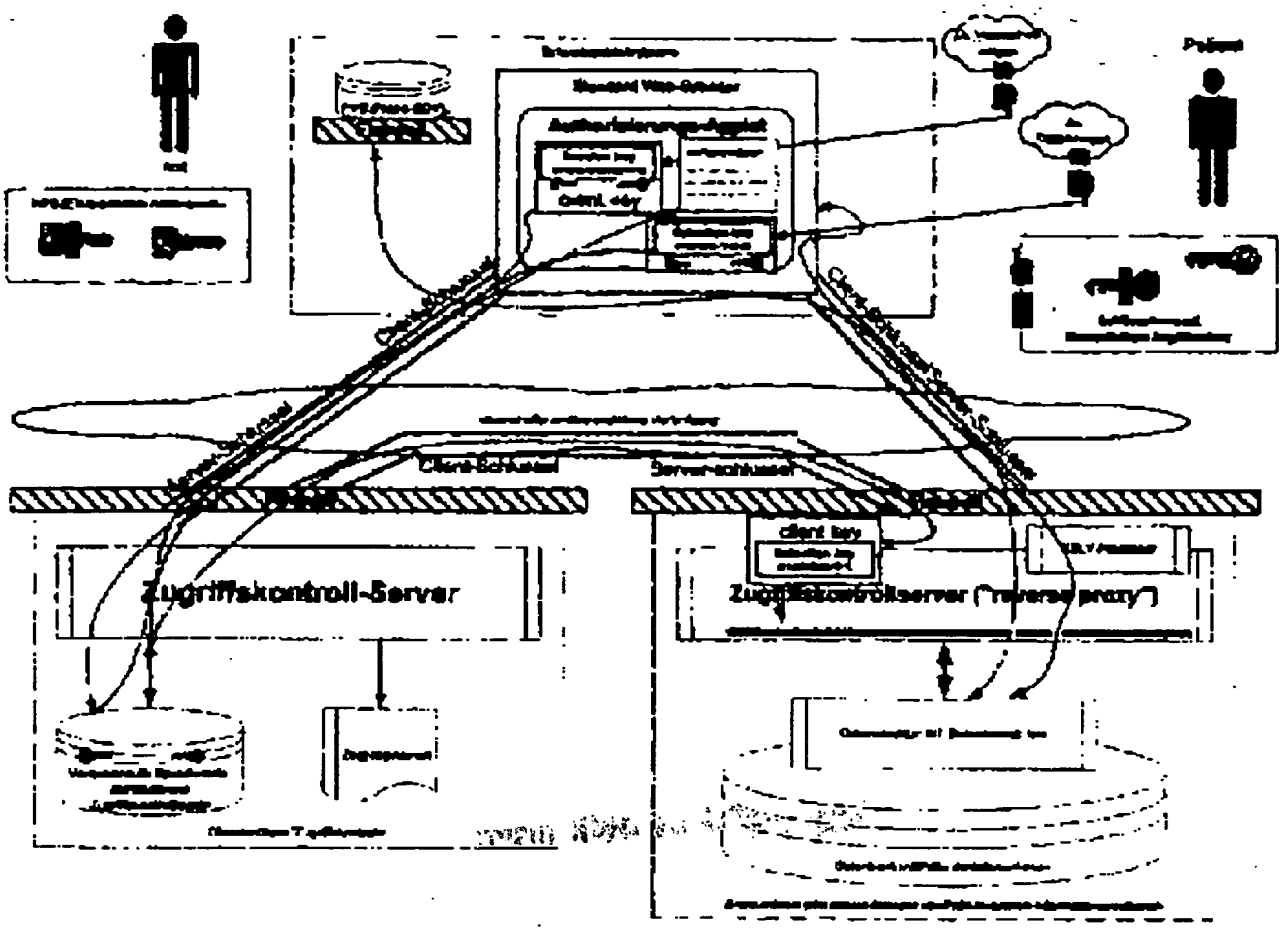
AN: PAT 2003-104212
TI: Method for context specific remote data access authentication. e.g. for controlling access to patient medical records by ensuring that both doctor and patient must be present to access patient records with their access chip cards
PN: **DE10121819**-A1
PD: 21.11.2002
AB: NOVELTY - Method for controlling access to electronically stored data in distributed heterogeneous environments in which data access is only granted when two or more persons are present to authorize the access. A third position for access testing is connected between data access and data storage positions for testing authorization of persons requiring access by use of public and private key cryptography. DETAILED DESCRIPTION - The inventive method relates to medical and other areas where encrypted data is used. The invention relates particularly to accessing of patient records. Thus both a patient and doctor must be present with their own access chip card between medical records can be accessed.; USE - Method for context specific remote data access authentication. ADVANTAGE - The inventive method ensures that data can only be accessed when the necessary security conditions are met, i.e. there are at least two persons with authentication means for accessing encrypted data. DESCRIPTION OF DRAWING(S) - (Drawing includes non-English language text). Figure shows a block diagram of an inventive method.
PA: (ROSN/) ROSNER W;
IN: ROSNER W;
FA: **DE10121819**-A1 21.11.2002;
CO: DE;
IC: G06F-012/14; H04L-009/28;
MC: T01-J06A1; T01-N02B1A;
DC: T01;
FN: 2003104212.gif
PR: DE1021819 04.05.2001;
FP: 21.11.2002
UP: 10.02.2003


Zugriffskontroll-Server

(19) BUNDESREPUBLIK DEUTSCHLAND

DEUTSCHES PATENT- UND MARKENAMT

(12) **Offenlegungsschrift**

(10) **DE 101 21 819 A 1**

(51) Int. Cl.[7]:
G 06 F 12/14
H 04 L 9/28

(21) Aktenzeichen: 101 21 819.2
(22) Anmeldetag: 4. 5. 2001
(43) Offenlegungstag: 21. 11. 2002

(71) Anmelder:
Rosner, Wolfgang, Dipl.-Ing. (FH), 95666 Mitterteich, DE

(74) Vertreter:
Anwaltskanzlei Paluka & Kollegen, 93047 Regensburg

(72) Erfinder:
gleich Anmelder

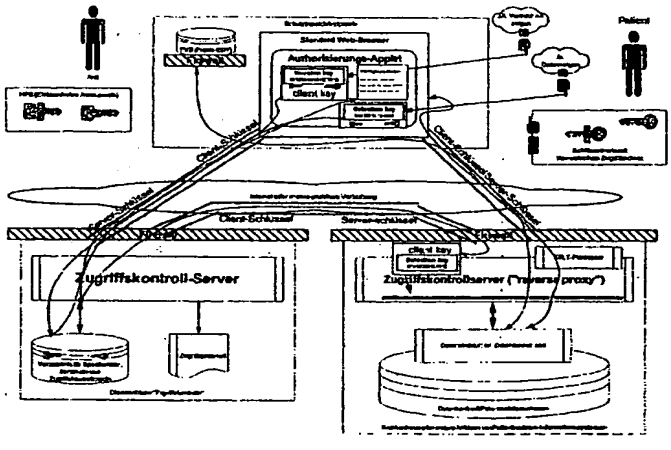
(56) Entgegenhaltungen:
DE 199 42 198 A1

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

Prüfungsantrag gem. § 44 PatG ist gestellt

(54) Verfahren zur kontextspezifischen Remote-Authentifizierung des Datenzugriffs

(57) Die bislang bekannt kryptografischen Verfahren erlauben mittels nur einer Chipkarte, die entweder vom Arzt oder vom Patienten geführt wird, entweder den Zugriff auf den gesamten Datenbestand des einzelnen Patienten oder auf Daten beliebig anderer Patienten.
Das neue Verfahren soll dem Arzt nur dann den Zugriff auf verschlüsselte personenbezogene gesundheitsrelevante und auch nur aus der Fachgruppe des für den Arzt relevanten Daten ermöglichen, wenn der Patient tatsächlich in der Praxis anwensend ist und durch Freigabe seiner Chipkarte sein Einverständnis erklärt.
Sowohl Arzt als auch Patient verfügen über eine kryptografische Chipkarte, die aufgrund der verfahrenseigenen Verschlüsselung der Übertragungswege bei Einführen beider Karten in ein Lesegerät an einem Rechner in der Arztpraxis die Authentifizierung eines Datenzugriffs durch mehrere Personen gewährleistet.
Das Verfahren eignet sich für den medizinischen und jeden Bereich, in dem datenschutzrelevante Daten zum Einsatz kommen.



DE 101 21 819 A 1

DE 101 21 819 A 1

Beschreibung

2.1 Technisches Gebiet

[0001] Zugriffskontrolle auf elektronisch gespeicherte, vertrauliche und schutzbedüftige Daten in verteilter und heterogener Umgebung, insbesondere auch im Internet

2.2 Stand der Technik

[0002] Die dargestellte Erfindung baut im wesentlichen auf der Technologie des Internets sowie kryptographischer Verfahren auf, also solcher, die digitale Verschlusselung und Signatur verwenden.

[0003] Soweit möglich, wird zum Stand der Technik auf die im Internet anerkannten RFC ("Request for Comments") Bezug genommen, die z. B. unter http:// / www.rfc-editor.org abrufbar sind. Diese RFC erfüllen einerseits die Funktion anerkannter Standards, andererseits dienen sie auch zur fachlichen Diskussion, so daß neue Entwicklungen hier frühzeitig ihren Niederschlag finden

2.2.1 Internet-Protokolle

2.2.1.1 TCP/IP

[0004] Das Rückgrat des Internets bildet eine standardisierte Gruppe von Netzwerk-Übertragungsprotokollen, die häufig unter dem Kürzel "TCP/IP" zusammengefaßt werden aber im weiteren Sinne auch beinhalten:

– Das "Internet-Protocol" (IP, [RFC 791]) weist jedem Rechner im Internet eine eindeutige Adresse aus 4 Bytes zu (typische Notation 123.234.56.78) Mehrere Rechner können zu Subnetzen zusammengefaßt werden.
Der Datentransfer erfolgt in Paketen (typ. ca 1 kByte groß) Schnittstellenrechner ("Router") übernehmen als Vermittlungsstellen die Verbindung zwischen Subnetzen. Muß ein Paket über mehrere Subnetze hinweg transportiert werden, wird es von Router zu Router weitergereicht. Im Normalfall ist davon auszugehen, daß weder Sender noch Empfänger die Kontrolle über alle Router und Subnetze der Übertragungsstrecke besitzen. Deswegen gilt das Internet als inhärent unsicher und anfällig für Abhören und Verfälschen von Daten sowie falsche Identitätsbehauptungen ("IP-Spoofing")
Eine neue Version des Internet-Protokolls (IP-V6, siehe [RFC 1883]) ist in Vorbereitung, hat sich aber noch nicht in der Praxis etabliert.
– Das "Transmission Control Protocol" (TCP, [RFC 793]) setzt auf IP auf und verdeckt die Paketübertragung vor den Anwenderprogrammen. Eine Anwendung fordert von TCP eine Verbindung zu einem bestimmten Rechner (identifiziert über die IP-Adresse) an. TCP stellt der Applikationen "Sockets" zur Verfügung, ein Schnittstellenpaar, über die sequentiell Daten gelesen und geschrieben werden können. Diese Sockets korrespondieren mit denen an der Gegenstelle, so daß die Anwendungen auf beiden Rechnern über dieses Socket-Paar kommunizieren können, ohne sich um die darunterliegende Netzwerkstruktur zu kümmern.
Um an einem Rechner verschiedene Programme (oder mehrere Instanzen des selben Programmes) unabhängig voneinander Verbindungen aufbauen zu lassen, beinhaltet TCP eine "Port-Adresse", die einen bestimmten Socket von ggf. mehreren auf einem Rechner identifizieren kann.
Das "User Datagramm Protocol" (UDP, [RFC 768]) ist eine etwas einfachere Alternative zu TCP und setzt ebenfalls auf IP auf. UDP kennt keine Verbindungen wie TCP, sondern sendet beliebig lange Datensequenzen ("Datagramme") an die Gegenstelle, ohne auf eine Bestätigung zu warten. UDP verdeckt damit vor allem die Vermittlungsschicht und die Paketstruktur.
Auch UDP kennt, wie TCP, mehrere Ports auf einem Rechner.
– Das Domain-Name-System (DNS, siehe [RFC 1034)/[RFC 1035]) ermöglicht die Zuordnung von Netzwerkadressen zu leicht einprägsamen Namen (typische Notation: host.organisation.land) und die Überlagerung einer organisatorisch ausgerichteten Gliederung des Netzes über die technische Struktur der IP-Adressen mit den Sub-Netzen.
Das DNS-System ist als verteilte Datenbank mit vielen Zwischenspeichern ausgelegt. Damit wird es (bei korrekter Konfiguration) sehr stabil, jedoch anfällig für Mißbrauch durch bewußte Fehlkonfiguration ("DNS-Spoofing") in begrenzten Netzsegmenten.

2.2.1.2 Universal Ressource Locators

[0005] Universal Ressource Locators (URL [RFC 1738]) ordnen jeder Netzwerkressource (z. B. Rechner, Mail-Konto, Textdatei, Bilddatei, Tondatei) einen String zu, über den die Ressource im Internet erreicht werden kann. URL enthalten typischerweise das verwendete Applikationsprotokoll, den anzusprechenden Server und ggf. weitere Informationen zur Spezifikation der Ressource relativ zum Server.

[0006] Das Konzept der URL wurde im aktuell gültigen Standard zum "Universal Ressource Identifier" (URI, [RFC 2396]) erweitert, das über die Netzerreichbarkeit hinaus Ressourcen eindeutig identifiziert.

[0007] Eine URL kann auch ein Programm oder Program-Schnittstelle identifizieren. In diesem Fall kann sie eine "query" enthalten, also eine Einheit von Daten, die vom identifizierten Programm interpretiert wird.

2.2.1.3 http

[0008] HTTP ("Hypertext Transport Protocol") setzt als Anwendungsprotokoll auf TCP auf. Aktuell im Einsatz sind die Versionen http 1.0 ([RFC 1945]) und http 1.1 ([RFC 2616]).

[0009] HTTP verbindet einen Client (typischerweise einen WWW-Browser) mit einem Server (einen Web-Server). Der Client sendet eine Anfrage an den Server, mit der er um die Übermittlung von Daten bittet. Sowohl der angesprochene Server als auch die Identifikation der Daten auf dem Server erfolgt über eine URL.

[0010] http erfolgt in isolierten "Request-Response"-Zyklen, zwischen denen keine Verbindung zwischen Client und Server aufrecht erhalten wird. Folgende Request-Arten sind gebräuchlich:

- GET: Der Client bittet um die Übermittlung einer Datei.
  Falls die adressierte Ressource keine Datei, sondern ein Programm ist, kann ergänzend noch eine "query", also eine Datensequenz, mitgereicht werden. In diesem Fall erstellt das aufgerufene Proramm eine Antwortdatei, die im Response zurückgereicht wird.
- POST: Der Client übermittelt an den Server eine Liste von Variablen mit Werten. POST-Requests sprechen fast immer Programme auf der Gegenseite an. POST-Requests resultieren typischerweise aus ausgefüllten HTML-Formularen.

[0011] Weitere Request-Arten (PUT, DELETE. . .) sind dagegen wenig gebräuchlich.

[0012] Das HTTP-Protokoll ermöglicht eine Authentifizierung des Benutzers (d. h. des Browsers gegenüber dem Server) mit Benutzernamen und Passwort.

[0013] Um http mit Status-Informationen zu erweitern, d. h. dafür zu sorgen, daß ein Server sich an Details eines vorhergehenden Requests des gleichen Clients "erinnern" kann, gibt es zwei gängige Mechanismen. Damit kann das an sich statuslose Protokoll das Konzept einer "Session", also einer Reihe von Request-Response-Zyklen mit inhaltlichem Zusammenhang, implementieren:

- Alle Seiten mit Hyperlinks innerhalb einer Session werden dynamisch generiert.
  Die Session-Variablen werden dabei an die Query aller internen Hyperlink-URLs angehängt (für GET-Requests) bzw. als POST-Variablen, z. B. in versteckten Formularfeldern, hinterlegt. Beim Aktivieren der Links werden diese Daten dann vom Browser wieder mit an den Server zurückgesandt.
- Eine Erweiterung des Standards ([RFC 2965]-"Cookies") ermöglicht es dem Server, im Response ein Variablen-Wert-Paar an den Browser zu senden, das dieser lokal speichert und dann bei jedem Request an den selben Server mit übermittelt.

2.2.1.4 HTML, XML und Hyperlinks

[0014] Die "Hypertext Markup Language" HTML wurde zunächst bis zur Version HTML 2.0 ([RFC 1866]) im Rahmen der Internetstandards definiert. Inzwischen (mit [RFC 2854]) wurde die Definition an das w3-Konsortium übertragen und weiterentwickelt ([HTML401], [XHTML1], [XML]).

[0015] HTML ermöglicht die formatierte Darstellung von Text (z. B. Schriftgröße, Absatzformate, Tabellen usw.) sowie das Einbinden beliebiger anderer Dateiformate (z. B. Bilder, Sound, Videos).

[0016] Darüberhinaus besitzt HTML "Hyperlinks", also Sprungverweise zu anderen Seiten, die im HTML-Quelltext mit ihrer URL definiert sind.

[0017] HTML und HTTP zusammen bilden das WWW (World Wide Web). Durch die in die Seiten eingebetteten Hyperlinks erscheint dem Benutzer der Eindruck, er würde auf einer Gruppe von Seiten "sich befinden", obwohl das HTTP-Protokoll keinen Verbindungsstatus kennt. Da die URL jeden beliebigen WWW-Server referenzieren können, erlaubt die geschickte Kombination von HTML-Seiten das Erstellen von "Applikationen", die sich über mehrere Server, möglicherweise weltweit verteilt, erstrecken. Durch die Einbindung von Programmen, die auch (über HTTP-POST oder eine query in HTTP-GET) Benutzereingaben annehmen, läßt sich die Funktionalität dieser "Web-Applikationen" beliebig erweitern und viele gängige Benutzeroberflächen einer Software funktional nachbilden.

2.2.1.5 ssl und tls

[0018] Die Datenübertragung in TCP/IP und damit auch in HTTP erfolgt grundsätzlich unverschlüsselt und mit begrenzter Fehlerkontrolle. Ein böswilliger Angreifer, der Zugang zu geeigneten Routern hat, kann den Datenverkehr abhören und verfälschen.

[0019] Um dies zu verhindern, hat die Firma Netscape eine zusätzliche Protokollebene standardisiert, die als ssl (siehe [SSL2]) bekannt ist. ssl setzt auf TCP auf und stellt der Applikation wiederum Sockets zur Verfügung (ist also für die Applikation während der Übertragung wieder transparent), ermöglicht aber die Absicherung der Übertragung mit verschiedenen kryptographischen Methoden (symmetrische Schlüssel, asymetrische mit ein- oder beidseitiger Authentifizierung) wie sie weiter unten beschrieben sind.

[0020] SSL wurde in einer Erweiterung als TLS ([RFC 2246]) in die Internet-Standards aufgenommen.

2.2.1.6 https

[0021] https, auch S-HTTP genannt, ist in [RFC 2660] spezifiziert und stellt die Implementierung des http-Protokolls über ssl- bzw. tls-Verbindungen dar. Algorithmen und Schnittstellen zur kryptographischen Infrastruktur sind in den gän-

gigen Web-Servern- und -Browsern implementiert.

2.2.1.7 Firewall

[0022] Da das Internet als unschützbarer, öffentlicher Raum gilt, müssen die Zugänge zu Netzen, die vertrauenswürdige Daten enthalten, entsprechend vor unberechtigtem Eindringen geschützt werden – vergleichbar der Pforte an einem Werkseingang. Die entsprechenden Vorrichtungen bezeichnet man als "Firewall". Einen Überblick über Techniken und Begriffe vermittelt [RFC 2647].

2.2.1.7.1 Firewall-Prinzipien

[0023] Firewalls verwenden zwei Grundprinzipien:

– Paketfilter: Es handelt sich hier um spezielle Router (vgl. Abschnitt TCP/IP), die zwischen dem externen und dem internen Subnetz stehen und nur nach bestimmten Regeln, ausgehend von Quell- und Zieladressen der Pakete, eine Verbindung herstellen, abweisen oder umleiten.
Paketfilter können meist auch IP-Adressen umwandeln ("Network address translation" NAT oder "Masquerading").
– Application-Level-Filter: Diese Filter treten gegenüber den Kommunikationspartnern als Gegenstelle aus und werten den Inhalt der übertragenen Daten aus (z. B. Datentypen, Inhalt, Viren usw.). Je nach Ergebnis der Auswertung wird der Inhalt dann transparent übertragen, gesperrt, verändert oder auch nur der Transfervorgang protokolliert.
Neben Sicherheitsüberprüfungen können diese Filter auch Daten zwischenspeichern und werden dann als "Proxy"-Server bezeichnet, wobei der Begriff auch synonym für Application level filter verwendet wird.
Application level filter müssen das jeweilige Protokoll (z. B. Mail, ftp, http. . .) verstehen, um den Inhalt analysieren zu können.

2.2.1.7.2 Firewall-Architekturen

[0024] Die Firewalls-Komponenten werden zu "Firewall-Architekturen" kombiniert. Gängige Grundkonfigurationen sind:
Einfacher Paketfilter:

(Internet)-----<Paketfilter>-----(internes Netz)

Routing findet nur nach festgelegten Regeln statt.
Es findet keine Inhaltsüberprüfung statt.
Einfacher Proxy:

(Internet)-----<Proxy>-----(internes Netz)

Es findet kein direktes Routing zwischen internem Netz und Internet statt.
Nur Protokolle, die im Proxy implementiert sind, können übertragen werden.
Dreigliedrige Firewall:

```
(Internet)------<Paketfilter>-------(internes Netz)
                      |
                   (DMZ)
                     +--<Proxy>
                     +--<Web-, Mail-, ... -Server>
```

Zweistufige Firewall:

```
(Internet)---<Paketf.1>--+--<Paketf.2>--(internes Netz)
                         |
                      (DMZ)
                        +--<Proxy>
                        +--<Web-, Mail-, ... -Server>
```

[0025] Bei der dreigliedrigen und der zweistufigen Firewall wird zwischen Internet und internem Netz eine "Entmilitarisierte Zone" (engl. demilitarized Zone = DMZ) als eigenes Subnetz – gleichsam als Sicherheitsschleuse – geschaffen.
[0026] Innerhalb der DMZ gelten niedrigere Sicherheitsanforderungen als im internen Netz, viele unberechtigte Zugriffe aus dem Internet werden jedoch bereits von der ersten Paketfilterstufe abgefangen.
[0027] In der DMZ befinden sich typischerweise die Proxy-Server zur Steuerung des Zugriffs zum internen Netz sowie ggf. die öffentlichen Server der Organisation.

2.2.2 Kryptographie

[0028] Die digitale Kryptographie löst unter Anwendung mathematischer (meist zahlentheoretischer) Verfahren verschiedene Sicherheitsanforderungen der digitalen Kommunikation während der Speicherung und/oder Übertragung, ins-

besondere

- Schutz vor unberechtigtem Zugriff von Daten durch Dritte
- Schutz vor unberechtigter Änderung von Daten
- Identifikation und Authentifizierung von Sender und Empfänger
- überprüfbare Verbindlichkeit und Unwiderruflichkeit von elektronisch abgegebenen Erklärungen

[0029] In der vorliegenden Erfindung werden verschiedene etablierte, im Folgenden erläuterte, Verfahren kombiniert.
[0030] Auf die tatsächlich erreichbare Sicherheit der Verfahren sowie deren Einflußparameter (Schlüssellänge, organsiatorische Maßnahmen, physikalische Sicherheit usw.) wird hier nur insofern eingegangen, als sie zum Verständnis der Erfindung relevant sind. Alle Erläuterungen beziehen sich auf die Annahme nicht korrumpierter Systeme.

2.2.2.1 symmetrische Verschlüsselung

[0031] Die einfachste Form der verschlüsselten Datenübertragung verwendet zur Ver- und Entschlüsselung die gleiche Binärsequenz als Schlüssel. Gängige Verfahren sind in [DES] und in [3DES] beschrieben. Eine verschlüsselte Nachricht kann nur von jemandem, der über den gemeinsamen Schlüssel verfügt, gelesen werden.
[0032] Wollen mehrere Parteien vertraulich miteinander kommunizieren, so besteht entweder (bei Verwendung eines gemeinsamen Schlüssels) nur ein gemeinsamer vertraulicher Kommunikationsraum oder es muß für alle Kommunikationspaare ein eigener Schlüssel vereinbart werden.

2.2.2.2 Asymmetrische Verschlüsselung

[0033] Die symmetrische Verschlüsselung erfordert eine vorab vorhandene vertrauliche Verbindung, um den Schlüssel auszutauschen. Dies kann in der typischen elektronischen Kommunikation häufig nicht gewährleistet werden. Diese Einschränkung umgehen asymmetrische Verfahren. Dabei verfügt jede Partei über einen Teil des Gesamtschlüssels, beide Teile müssen (nach spezifischen Kriterien des Algorithmus) zusammenpassen ("Schlüssel-Schloß-Prinzip").
[0034] Eine Nachricht, die mit einem Schlüsselteil verschlüsselt wird, kann nur mit dem jeweils passenden Teil des Schlüsselpaares entschlüsselt werden. Die Reihenfolge, in der die Schlüssel angewendet werden, ist dabei (zumindest bei RSA) gleichgültig, ebenso können mehrere Schlüsselpaare nacheinander angewendet werden.

2.2.2.2.1 Algorithmen

[0035] Bekannte Algorithmen für asymmetrische Verschlüsselung sind Diffie-Hellman ([RFC 2631] [US-Pat No. 4,200,770]) und RSA ([RSA][RFC 2313] (als PKCS #1); [US-Pat No. 4,405,829]), DSA sowie Algorithmen auf der Basis Elliptischer Kurven ([PKCS13]).
[0036] Soweit nicht anders angegeben, beziehen sich die folgenden Ausführung auf RSA, sind aber weitgehend auf andere Algorithmen übertragbar.

2.2.2.2.2 Public-Private-Key-Verfahren

[0037] Bei diesem Verfahren (oft auch mit PKI – Pubfic Key Infrastructure – referenziert) behält jeder Teilnehmer an einem Kommunikationssystem einen Schlüssel des Paares, während der andere an alle anderen Teilnehmer veröffentlicht wird.

2.2.2.2.2.1 Verschlüsseln

[0038] Zum Verschlüsseln einer Nachricht gegen unberechtigtes Lesen verschlüsselt der Sender mit dem ihm bekannten öffentlichen Schlüssel des Empfängers. Nur dieser verfügt über den dazu passenden privaten Schlüssel und kann die Nachricht entziffern.

2.2.2.2.2.2 Signatur (digitale Unterschrift)

[0039] Der Sender verschlüsselt seine Nachricht mit seinem eigenen privaten Schlüssel. Jeder im System kann diese Nachricht lesen, da der dazugehörige Schlüssel im Paar per Definition öffentlich ist.
[0040] Das Lesen mit dem öffentlichen Schlüssel des Senders gelingt jedoch nur, wenn tatsächlich der private Schlüssel des behaupteten Senders zum Verschlüsseln verwendet wurde. Ein böswilliger Dritter, der sich mißbräuchlich als Sender ausgeben möchte, aber nicht über dessen privaten Schlüssel verfügt, kann dies nicht erreichen. Damit ist die Authentizität des Senders gesichert.
[0041] Der Sender kann eine einmal abgegebene Nachricht nicht verleugnen, wenn sie entziffert werden konnte, da nur er in der Lage war, diese Nachricht zu erzeugen.

2.2.2.2.2.3 Trust-Center

[0042] Public-Key-Systeme erfordern, daß jeder Kommunikationspartner sich auf die Echtheit der von ihm verwendeten öffentlichen Schlüssel verlassen kann.
[0043] Im einfachen Fall kann das durch einen vorherigen Austausch mit dem Inhaber über einen sicheren Kanal erfolgen. Der erforderliche Austauschaufwand steigt jedoch mit dem Quadrat der Teilnehmer und wird deswegen in grö-

ßeren Systemen zu aufwendig.

[0044] Für diesen Fall kann eine "Vertrauenswürdige Stelle" (häufige Synonyme: Zertifzierungsstelle, Netznotar, Certificate Authority, Trust-Center) zwischengeschaltet werden. Der einzelne Nutzer tauscht nur mit dieser Stelle die öffentlichen Schlüssel über einen sicheren Kanal aus. Das Trust-Center "signiert" den öffentlichen Schlüssel der Benutzer und stellt so deren Integrität sicher.

[0045] Die Verteilung der signierten öffentlichen Schlüssel ("Zertifikate") kann durch den Benutzer selbst – z. B. mit einer Nachricht oder durch Veröffentlichung im Nutzerkreis – erfolgen.

[0046] Trust-Center-Zertifikate können auch kaskadiert werden ("Certificate Chain"), d. h. die Integritätsprüfung erfolgt indirekt über ein Center, das dem betreffenden Nutzer gegenüber nicht direkt, sondern über ein weiteres Trust Center authentifiziert wurde. Analog können zwei zunächst unabhängige Trust-Center sich gegenseitig zertifizieren (Cross Certificate) und so die Benutzerkreise vereinigen.

[0047] Einzelheiten des üblichen (standardisierten) Verfahrens sind in [X.509] beschrieben.

2.2.2.2.2.4 Verzeichnis-Dienste

[0048] Die Veröffentlichung der X-509-Schlüssel kann in sogenannten "Verzeichnissen" nach [X.500] erfolgen, wie sie speziell für das Internet in [RFC 2459] und [RFC 2510] beschrieben sind.

[0049] Das gängige Protokoll hierfür ist LDAP [RFC 1777], insbesondere in der aktuellen Version LDAP v3 [RFC 2251].

[0050] Diese Verzeichnisse sind als eigenständige Serverdienste angelegt. Sie ermöglichen die Suche eines Kommunikationspartners nach verschiedenen Kriterien (unter anderem auch eines weltweit eindeutigen, hierarchischen Namens nach X.500-Konvention, auch als DN = "distinguished Name" bezeichnet). Die Hinterlegung von X.509-Zertifikaten in LDAP-Verzeichnissen ist Bestandteil der Spezifikation.

2.2.2.2.2.5 Smart-Cards

[0051] Empfindlichste Angriffsstelle für ein PKI ist der Zugriff Unbefugter auf einen privaten Schlüssel. Da es sich dabei nur um eine binäre Sequenz handelt, kann dieser Schlüssel unbemerkt kopiert und von einem Angreifer verwendet werden (PC-Wartung, Verwendung in einem fremden Rechner, bösartige Software auf einem PC. . .).

[0052] Um dies zu verhindern, können die privaten Schlüssel auf unabhängigen Geräten gespeichert werden, die die erforderlichen Verschlüsselungsoperationen durchführen, ohne daß der private Schlüssel das Gerät verläßt, und die konstruktiv überhaupt nicht in der Lage sind, den privaten Schlüssel auszugeben. Unter verschiedenen vorgeschlagenen Bauformen hat sich hierzu die "Smart-Card" im Kreditkartenformat etabliert (siehe [ISO 7816]-15).

[0053] Das Datenblatt eines Beispielproduktes findet sich unter [SecurID 3100].

2.2.2.2.2.6 Biometrische Identifizierung

[0054] Smart-Cards und ähnliche "Security Tokens" können entwendet oder vom Inhaber im guten Glauben leichtfertig überlassen werden. Biometrische Verfahren (z. B. Fingerabdruck, Iris-Muster) ermöglichen dagegen die eindeutige Identifizierung einer Person direkt anhand von Körpermerkmalen.

[0055] Allerdings ist die Zuverlässigkeit biometrischer Verfahren von der Integrität der prüfenden Systeme am Ort der zu identifizierenden Person abhängig (vgl. [Biometrics]). In der Remote-Authentifizierung kann ein Angreifer durch ein fingiertes System die Reaktion eines echten Systems vortäuschen, ohne daß die zu identifizierende Person tatsächlich anwesend ist.

[0056] Es bietet sich jedoch an, die Verwendung privater Schlüssel auf Smart-Cards (oder anderen Tokens) mit biometrischer Zugriffskontrolle zu sichern, so daß vor jeder Verwendung des privaten Schlüssels ein willentlicher Akt des Inhabers vorausgehen muß (Vergleiche auch [ISO 7816]-11 bzw. zugehörige Normierungsentwürfe).

2.2.2.3 Hybrid-Verfahren

[0057] Asymmetrische Verfahren erfordern aufgrund ihrer Komplexität wesentlich höheren Rechenaufwand als symmetrische Verfahren. Dies wird durch die Verwendung von (meist sehr minimalistischen) Tokens wie Smart-Cards weiter verschärft, die im Gegensatz zu modernen Prozessoren in Anwendungsrechnern nur einen sehr begrenzten Datendurchsatz aufweisen.

[0058] In vielen Einsatzgebieten werden deshalb verschiedene Verfahren kombiniert, um den Datendurchsatz zu optimieren.

2.2.2.3.1 Session Keys

[0059] Ein "Session Key" ist eine Zufallszahl, mit der unter Anwendung eines symmetrischen Verfahrens der Großteil des Kommunikationsinhaltes verschlüsselt wird.

[0060] Das asymmetrische Verfahren (und ggf. das minimalistische Token) braucht dann nur mehr den Session Key verschlüsseln. Dieser verschlüsselte Session Key wird zusammen mit der (symmetrisch) verschlüsselten Nachricht übertragen. Der Empfänger entschlüsselt zunächst den Session Key und mit diesem dann den Rest der Nachricht.

2.2.2.3.2 Hash-Funktionen

[0061] Hash-Funktionen sind sogenannte Einweg-Funktionen, die aus einer längeren Datensegenz einen (im statisti-

schen Rahmen) vergleichsweise kurzen, aber eindeutigen Wert (vergleichbar einer Prüfsumme) ermitteln, der oft auch als "Fingerprint" oder "Hash-Wert" bezeichnet wird. Umgekehrt ist es jedoch nicht möglich, aus der "Prüfsumme" auf die Daten selber zu schließen.

[0062] Gängige Algorithmen sind bekannt unter den Namen MD2, MD4, MD5 (siehe hierzu [RFC 1321]) oder [SHA]. Die Anwendung von Hash-Funktionen findet sich etwa als HMAC unter [RFC 2104].

2.2.2.3.3 E-Mail und S-MIME

[0063] Das Verschlüsseln und Signieren von elektronischen Nachrichten als E-Mail ist weit verbreitet. Alle gängigen e-Mail-Programme verfügen über die entsprechende Technologie. Die Technologie und Infrastruktur ist ausgereift und kann als "Musterlösung" für die Implementierung komlexer kryptographischer Verfahren gelten.

[0064] Die Details des Standards "S-MIME" für verschlüsselte Übertragung von e-Mails im Internet sind in [RFC 1421], [RFC 1422], [RFC 1423] und [RFC 1424] festgelegt.

2.2.2.4 Authentifizierungs-Token

[0065] In allen verteilten Rechnersystemen (meist Client-Server-Systeme) ist eine Authentifizierung des Benutzers am Zielrechner erforderlich. Im einfachen Fall authentifiziert sich der Benutzer (typischerweise mit Benutzername und Passwort) an jedem Zielsystem einzeln. Bei größeren Systemen führt das zu einem erheblichen Administrationsaufwand im System und zur Verwirrung der Benutzer mit vielen Passwörtern.

[0066] Authentifizierungsverfahren mit einem eigenständigen Prüfsystem sind seit längerem bekannt. Bei diesen Verfahren existiert neben der datenanfordernden Stelle (z. B einem Arbeitsplatzrechner, also dem Client) und der datenhaltenden Stelle (z. B. einem Fileserver) ein eigener Authentifizierungsrechner. Der Client meldet sich zunächst am Authentifizierungsrechner an, nur dieser muß in der Lage sein, die Identitätspüfung für jeden Nutzer vorzunehmen. Er erstellt dann ein "Token" oder "Ticket" nach kryptographischen Methoden, das an den Client zurückübermittelt wird. Mit diesem Token kann der Client sich dann an verschiedenen Datenservern authentifizieren.

2.2.2.4.1 Kerberos

[0067] Als Prototyp der Token-basierten Authentifizierung gilt das am MIT entwickelte Kerberos, das in seiner aktuellen Version V5 als Internet-Standard ([RFC 1510]) etabliert ist. Eine Kerberos-Variante wird unter anderem auch im Microsoft-Betriebssystem WINDOWS 2000 eingesetzt und löst dort einen proprietären Mechanismus, wie er in NT 4 verwendt wurde, ab.

[0068] Kerberos verwendet symmetrische Schlüssel für jeden Benutzer, die in den Authentifizierungsrechnern abgelegt sind.

2.3 Problemstellung

2.3.1 Konkrete Aufgabenstellung im medizinischen Umfeld

[0069] Ein Arzt behandelt einen Patienten und braucht dazu Daten (z. B. Befund, Röntgenbild. . .), die von einem Dritten (z. B. einem Krankenhaus, einem Labor oder einer anderen Arztpraxis) – "Datenhalter" genannt – gespeichert werden.

[0070] Verschiedene Rechtsvorschriften, Berufstandsregeln, Vertrauenserwartungen der Patienten, Vertrauenskonstellationen usw. verpflichten den Datenhalter, nur berechtigten Personen einen Zugriff auf definierte und von den jeweiligen Personen abhängigen Bereich der Daten zu gewähren.

[0071] Erforderliche Informationen, die der Datenhalter braucht, um Zugriffsentscheidungen zu treffen, sind etwa:

- Ist die anfordernde Person tatsächlich Arzt und wenn ja, in welchem Fachgebiet?
- Ist der Patient, den die Daten betreffen, anwesend und hat er seine Zustimmung erteilt?
- Ist sich der Patient über die gespeicherten Daten im Einzelnen bewußt (will er z. B. einem Arzt, der ihn im Auftrag einer Versicherung untersucht, offenbaren, daß es umfangreiche Daten aus einer Herzklinik gibt)?
- Gibt es eine besondere Vertrauensstellung zwischen der anfordernden Stelle und dem Datenhalter, z. B. aufgrund besonderer vertraglicher Regelungen?

[0072] Die Anwendung der gebotenen hochsicheren Übertragungsverfahren nach dem Stand der Technik überfordert regelmäßig die beteiligten Parteien, so daß die Absicherung der Übertragung an Dritte ("sicherheitsvermittelnde Stelle") auszulagern ist, die jedoch keine Berechtigung haben, die Daten selber zu lesen.

[0073] Kosten- und Effizienzkriterien erfordern, daß auf Seiten des Datenhalters und der sicherherheitsvermittelnden Stelle die Abwicklung ohne direkte Einflußnahme von Menschen erfolgt.

[0074] Neben der reinen Authentifizierung der beteiligten Personen können noch weitere Informationen des Anforderungskontextes erforderlich werden, die ggf. iterativ abgefragt werden müssen.

2.3.2 Verallgemeinerung der Problemstellung

[0075] Die dargestellte Lösung betrifft die Verschlüsselung der Übertragungswege und beinhaltet keine Einschränkung der Daten selber, deswegen kann der Einsatz auch außerhalb des medizinischen Umfeldes erfolgen.

[0076] Damit kann die Aufgabenstellung erweitert werden:

Es soll einem datenverarbeitenden System ermöglicht werden, bei einem angeforderten Datenzugriff Informationen über den Kontext des beabsichtigten Zugriffs, insbesondere mehrerer beteiligter Personen, des Zeitpunktes, und ggf. weiterer anwendungsspezifischer Informationen zu erhalten. Der Betrieb des Zugriffskontrollsystems soll dabei von der Datenhaltung entkoppelt werden können.

2.4 Wesen der Erfindung

[0077] Es werden die bekannten Sicherheits- und Authentifizierungsverfahren dergestalt kombiniert, daß zur Authentifizierung eines Datenzugriffs mehrere Personen zustimmen müssen, weitere Informationen aufgenommen werden können und die Rollen verschiedener Schritte im Authentifizierungsprozess räumlich und datentechnisch weitgehend entkoppelt sind.

[0078] Damit wird es möglich, die Verteilung der Verantwortung für Teilbereiche örtlich und organisatorisch zu trennen und Systeme auf unterschiedlichen Plattformen zu integrieren.

2.5 Gewerbliche Anwendbarkeit

[0079] Alleine im Gesundheitswesen wird von Fachleuten ein enormes Rationalisierungspotential sowie eine Qualitäts- und Serviceverbesserung vorhergesagt, wenn die Kommunikation zwischen mehreren Ärzten, die einen Patienten behandeln, verbessert wird.

[0080] Die elektronische Kommunikation, wie sie in vielen anderen Bereichen heute bereits üblich ist, könnte dafür einen wichtigen Beitrag leisten, scheiterte bisher aber an einer Infrastruktur, die einerseits die erforderlichen Sicherheitsanforderungen erfüllt und andererseits für die betroffenen Akteure handhabbar bleibt.

[0081] Die beschriebene Erfindung kann als Grundlage für diese Sicherheits-Infrastruktur dienen.

[0082] Weitere Anwendungsfälle ergeben sich z. B. in der Rechtspflege, der Wirtschafts- und Steuerberatung, der Personalvermittlung und -beratung, dem Versicherungswesen, dem Handel mit Käuferprofilen oder dem Finanzwesen.

2.6 Vorteile

[0083] Bisherige Lösungen auf bekannten Standards ermöglichen immer nur die Authentifizierung einer Person (z. B. des Arztes). Dieser hat dann Zugriff auf einen wesentlich größeren Datenbestand, nicht nur auf den jeweiligen Patienten, was hier eingeschränkt werden kann. Durch die Übertragung weiterer Kontextdaten können beliebig komplexe Sicherheitsphilosophien implementiert werden.

[0084] Ein weiterer Vorteil der beschriebenen Lösung ist die Modularität. Es können damit einfach bestehende Systeme verschiedener Datenhalter und Datenverwender mit einfachen Schnittstellen kombiniert werden zu einem sicheren Datenaustauschnetzwerk.

2.7 Ausführungsbeispiel

siehe Abschnitt 3

2.8 Literaturnachweis

zum Stand der Technik:

[3DES]
American National Standards Institute
Triple Data Encryption Algorithm Modes of Operation.
ANSI X9.52-1998; 1998

[Biometrics]
Gaël Hachez, François Koeune and Jean-Jacques Quisquater Biometrics, access control, smart cards: a not so simple combination
http:// www.dice.ucl.ac.be/crypto/publications/Biometrics.pdf

[DES]
American National Standards Institute
American National Standard for Information Systems – Data Link Encryption
ANSI X3.106; 1983

[HTML401]
Dave Raggett, Arnaud Le Hors, Ian Jacobs
HTML 4.01 Specification
W3C Recommendation; December 1999
http:// www.w3.org/TR/html401

[ISO 7816]
Gisela Meister
ISO/IEC 7816 Standards: Status and New Work Items Giesecke & Devrient, München
http:// sit.gmd.de/SICA/papers/WS_01/Beitrag_Meister

[PKCS12]
RSA Laboratories
PKCS #12: Personal Information Exchange Syntax Standard Version 1.0; June 24, 1999
http:// www.rsasecurity.com/rsalabs/pkcs/pkcs-12/index.html

[PKCS13]
RSA Laboratories
PKCS #13: Elliptic Curve Cryptography Standard PROJECT OVERVIEW January 12, 1998
http:// www.rsasecurity.com/rsalabs/pkcs/pkcs-13/project_overview.html
[RFC 768]
J. Postel – ISI
User Datagram Protocol
RFC 768; ISI, 28. August 1980
[RFC 791]
DARPA Internet Program Protocol Specification Internet Protocol
RFC 791; September 1981
[RFC 793]
DARPA Internet Program Protocol Specification
Transmission Control Protocol
RFC 793; September 1981
[RFC1034]
Mockapetris, P
Domain Names – Concepts and Facilities
RFC 1034, November 1987
[RFC1035]
Mockapetris, P.
Domain Names – Implementation and Specification"
RFC 1035, November 1987
[RFC 1321]
Rivest, R.
The MD5 Message-Digest Algorithm
RFC 1321; April 1992
[RFC 1421]
Linn, J.
Privacy Enhancement for Internet Electronic Mail:
Part I: Message Encryption and Authentication Procedures
RFC 1421; DEC, February 1993
[RFC 1422]
Kent, S.
Privacy Enhancement for Internet Electronic Mail:
Part II: Certificate-Based Key Management
RFC 1422; BBN, February 1993
[RFC 1423]
Balenson, D.
Privacy Enhancement for Internet Electronic Mail:
Part III: Algorithms, Modes, and Identifiers
RFC 1423; TIS, February 1993
[RFC 1424]
Balaski, B.
Privacy Enhancement for Internet Electronic Mail:
Part IV: Notary, Co-Issuer, CRL-Storing and CRL-Retrieving Services
RFC 1424; RSA Laboratories, February 1993
[RFC 1510] Kohl, J. and B. Neuman
The Kerberos Network Authentication System (V5)
RFC 1510; September 1993
[RFC 1738] Berners-Lee, T., Masinter, L., and M. McCahill
Uniform Resource Locators (URL)
RFC 1738; December 1994
[RFC 1777] Yeong, Y., Howes, T. and S. Kille
Lightweight Directory Access Protocol
RFC 1777; March 1995
[RFC 1866] Berners-Lee, T. and D. Connolly
Hypertext Markup Language – 2.0
RFC 1866; November 1995
[RFC 1883]
Deering, S. and R. Hinden
Internet Protocol, Version6 (IPv6) Specification
RFC 1883; December 1995
[RFC 1945]
Berners-Lee, T., Fielding, R. and H. Frystyk
Hypertext Transfer Protocol – HTTP/1.0
RFC 1945; May 1996
[RFC 2104]

Krawczyk, H., Bellare, M., and R. Canetti
HMAC: Keyed-Hashing for Message Authentication
RFC 2104; February 1997
[RFC 2246]
Dierks, T, and C. Allen
The TLS Protocol Version 1.0
RFC 2246; January 1999
[RFC 2251]
M. Wahl, T. Howes, S. Kille
Lightweight Directory Access Protocol (v3)
RFC 2251; December 1997
[RFC 2313]
Kaliski, B.
PKCS #1: RSA Encryption Version 1.5
RFC 2313; March 1998
[RFC 2396]
Berners-Lee, T., Fielding, R. and L. Masinter
Uniform Resource Identifiers (URI): Generic Syntax"
RFC 2396; August 1998
[RFC 2459]
Housley, R., Ford, W., Polk, W. and D. Solo
Internet X.509 Public Key Infrastructure Certificate and CRL Profile
RFC 2459; January 1999
[RFC 2510]
C. Adams, S. Farrell
Internet X.509 Public Key Infrastructure Certificate Management Protocols
RFC 2510; March 1999
[RFC 2616]
Fielding, R., Gettys, J., Mogul, J., Frysyk, H., Masinter, L. Leach, P. and T. Berners-Lee
Hypertext Transfer Protocol – HTTP/1.1
RFC 2616; June 1999
[RFC 2631]
Rescorla, E.
Diffie-Hellman Key Agreement Method
RFC 2631; June 1999
[RFC 2647]
D. Newman
Benchmarking Terminology for Firewall Performance
RFC 2647; August 1999
[RFC 2660]
E. Rescorla, A. Schiffman
The Secure HyperText Transfer Protocol
RFC 2660, August 1999
[RFC 2854]
D. Connolly, L. Masinter
The "text/html" Media Type
RFC 2854; June 2000
[RFC 2965]
D. Kristol, L. Montulli
HTTP State Management Mechanism
RFC 2965; October 2000
[RSA]
R. Rivest, A. Shamir, and L. M. Adleman
A Method for Obtaining Digital Signatures and Public-Key Cryptosystems Communications of the ACM, v. 21, n. 2, Feb 1978, pp. 120–126.
[SecurID 3100]
RSA SecurID® 3100 Smart Card
Secure RSA Cryptographic Container for PKI Credentials (Datasheet)
http:// www.rsasecurity.com
/products/securid/datasheets/ds3100smartcard.html
[SHA]
NIST
Secure Hash Standard
FIPS PUB 180-1; April 1995
[SSL2]
Hickman, Kipp
The SSL Protocol
Netscape Communications Corp., Feb 9, 1995

[US-Pat No. 4,200,770]
Cryptographic Apparatus and Method ("Diffie-Hellman")
[US-Pat No. 4,218,582]
Public Key Cryptographic Apparatus and Method ("Hellman-Merkle")
[US-Pat No. 4,405,829]
Cryptographic Communications System and Method ("RSA")
[US-Pat No. 4,424,414]
Exponential Cryptographic Apparatus and Method ("Hellman-Pohlig")
[X.500]
Information Technology – Open Systems Interconnection - The Directory: Overview of concepts, models and services
ITU-T Recommendation X.500
ISO/IEC 9594-1; 1997
[X.501]
Information Technology – Open Systems Interconnection The Directory: Models
ITU-T Recommendation X.501
ISO/IEC 9594-2; 1997
[X.509]
Information Technology – Open Systems Interconnection
The Directory: Authentication framework
ITU-T Recommendation X.509
ISO/IEC 9594-8; 1997
[X.520]
Information Technology – Open Systems Interconnection The Directory: Selected attribute types.
ITU-T Recommeridation X.520
ISO/IEC 9594-6; 1997
[X.680] Abstract Syntax Notation One (ASN.1) – Specification of Basic Notation
ITU-T Recommendation X.680; 1994
[XHTML1]
Steven Pemberton et. al.
XHTML 1.0: The Extensible HyperText Markup Language:
A Reformulation of HTML 4 in XML 1.0
W3C Recommendation, January 2000
http:// www.w3.org/TR/xhtml1
[XML]
Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler
Extensible Markup Language (XML) 1.0 (Second Edition)
W3C Recommendation; 6 October 2000
hftp:// www.w3.org/TR/REC-xml

3.1 Vorbemerkung

[0085] Das beschriebene Ausführungsbeispiel beruht auf der konkreten Aufgabenstellung im medizinischen Kontext (vgl. Erläuterungen zum Patentantrag) und verwendet deshalb die entsprechenden Begriffe "Arzt", "Patient" usw. Die zugrunde liegende Aufgabenstellung besteht darin, dem Arzt Zugriff auf vertrauliche Daten des Patienten, den er gerade behandelt, zu gewähren, und zwar nur zu diesem Patienten und nur in einem Umfang, wie es seiner standesrechtlichen Zulassung und der Einwilligung des Patienten entspricht (vgl. Anspruch 1).

[0086] Mögliche Erweiterungen und Varianten des Verfahrens sind teilweise (in Kursivdruck) im Text sowie im letzten Abschnitt beschrieben, ergeben sich aber insbesondere aus den Patentansprüchen.

3.2 Architektur

[0087] Die an einem Gesamtsystem beteiligten Komponenten sind in der Zeichnung 1 skizziert.

[0088] Es wird für die drei wesentlichen Bestandteile – Anforderungsstelle, Zugriffskontrolle und Datenhaltung – immer im Singular gesprochen. In einem ausgebauten System können natürlich von all diesen Stellen mehrere Instanzen auftreten, die auch miteinander weiter vernetzt (z. B. über HTML-Links) sein können.

3.2.1 Arbeitsstation zur Zugriffsanforderung

[0089] Am Arbeitsplatz des Arztes befindet sich ein PC mit einem Standard-Web-Browser. Es wird angenommen, daß der Zugriff auf die Daten vorzugsweise über HTML/http erfolgt, da damit eine einfache Integration und Standardisierung erreicht werden kann.

[0090] Ergänzend kann an der Arztpraxis auch ein eigenes Praxis-Managent-System ("PMS") vorliegen, das entweder über den Browser oder unabhängig, aber sinnvollerweise auch über http/https Daten nach Freigabe abrufen kann. Die Integration des PMS ist jedoch für das Verständnis der vorliegenden Erfindung unwesentlich.

[0091] Die Zuverlässigkeit des PCs und der Praxissoftware unterliegt der Kontrolle des Arztes.

3.2.2 Schlüsselkarten der Benutzer

[0092] Sowohl Arzt als auch Patient verfügen über eine Smart-Card mit dort abgelegtem privatem Schlüssel und integriertem Kryptoprozessor (nach Anspruch 7 und 8).

[0093] Für Ärzte gibt es dafür bereits einen etablierten Standard ("health professional card" – HPC), der diese Vorgaben erfüllt.

[0094] Für die Patienten gibt es keinen definierten Standard. Die Karte des Patienten sei mit einem Fingerabdruckleser ausgestattet, der vor jeder Verschlüsselungsaktion ausgelöst werden muß und dafür sorgt, daß vor jeder Schlüsselverwendung eine explizite, persönliche Willensäußerung des Patienten vorliegt (Anspruch 9 und 10).

[0095] Die Arbeitsstation des Arztes muß mit geeigneten Lesegeräten ausgestattet sein, um die Smart-Cards (gleichzeitig oder nacheinander) anzusteuern.

3.2.3 Zugriffskontrollsystem

[0096] Das Zugriffskontrollsystem ("Access Management Hub") befindet sich räumlich und organisatorisch unabhängig von Arztpraxis und Datenbank.

[0097] Es ist über eine Firewall mit dem Internet verbunden.

[0098] Es verfügt über eine Datenbank der öffentlichen Schlüssel aller registrierten Ärzte sowie aller Zertifizierungsstellen, die zur Registrierung von Patienten befugt sind (Access directory).

[0099] Diese Datenbank kann am Standort der Zugriffskontrollstelle selbst liegen oder über geeignete Directory-Protokolle (z. B. LDAP) ganz oder teilweise von anderen Standorten importiert werden (vgl. Anspruch 21 ff).

Erweiterung

[0100] Werden – abweichend vom Bild – mehrstufige Zertifikatsketten verwendet (vgl. Anspruch 18), reicht es für die prüfende Stelle aus, die Zertifikate der untersten Ebenen der verwendeten Ketten vorzuhalten. In diesem Fall müssen die vollständigen Zertifikate einschließlich ggf. aller Zwischenzertifikate auf den Smartcards der zu authentifizierenden Personen abgelegt sein und als Bestandteil des Authentifizierungsantrages mit übertragen werden (Anspruch 20).

[0101] Die Zugriffskontrollstelle stellt das empfindlichste System in der Sicherheitskette dar und befindet sich deshalb in einem eigenständigen hochsicheren Bereich eines spezialisierten Dienstleisters ("Access security provider").

Erweiterung

[0102] Alle Vorgänge können – z. B. zur Beweissicherung in Streitfällen – mitprotokolliert werden ("Access Log").

Erweiterung

[0103] Die Zugriffskontrollstelle kann bereits Informationen vorhalten, wo welche Daten über einen Patienten abgespeichert sind ("Location directory")

3.2.4 Datenspeicher

[0104] Die Datenbank mit Patientendaten wird als bestehend angenommen.

[0105] Die Schnittstelle für Fernzugriff sollte möglichst nahe an die Anforderung browsergestützter Systeme kommen, was beispielsweise durch die Anwendung der XML-Version des internationalen Patientendatenstandards "hl7" erreicht werden kann.

[0106] Der Datenspeicher ist aus dem Internet nicht direkt erreichbar (Anspruch 39).

3.2.5 Freigabesystem

[0107] Das Freigabesystem ("Zugriffskontrollserver") befindet sich hinter einer Firewall (Anspruch 39) am Standort des Datenhalters, z. B. eines Krankenhauses ("Hospital or other maintainer of health information services").

[0108] Im Beispiel wird angenommen, daß die Datenbank selber Informationen im XML-Format liefert, das von den gängigen heutigen Browsern nur eingeschränkt gelesen werden kann. In diesem Fall kann das Freigabesystem eine Übersetzung ("XSLT-Processor") in Standard-HTML vornehmen. Es können aber auch XML-Daten direkt durchgereicht werden (nach erfolgter Freigabe) oder, wenn die Datenbank selbst HTML-Format liefern kann, kann auch dieses nach Prüfung weitergereicht werden. Durch diese Variationsmöglichkeiten ergeben sich breite Anpassungsmöglichkeiten an vorhandene Systeme.

[0109] Für die Implementierung kann jede gängige Programmierumgebung verwendet werden. Java-Servlets bieten beispielsweise die erforderlichen HTTP-Schnittstellen und sind offen für die wichtigen Netzwerk-, Krypto- und XML-Bibliotheken.

[0110] Der Zugriff auf den Datenspeicher erfolgt typischerweise nochmals über eine Paketfilterung auf IP- und Port-Ebene (dicke schwarze Linie mit Löchern), so daß sich das Freigabesystem in einer DMZ zwischen zwei Paketfiltern befindet (vgl. "Stand der Technik/Firewall" sowie Anspruch 39 und 40). Alternativ, aber mit gleicher Funktionalität, erfolgt die Kommunikation zwischen Firewall und Datenbank über ein anderes Protokoll als TCP/IP, so daß ein direktes Routing aus dem Internet effizient verhindert wird.

[0111] Das Freigabesystem erhält seine Zugriffskontrollinformationen aus der Datenbank (Anspruch 30 ff, im Bild angedeutet mit dem dicken Doppelpfeil), wobei die genaue Implementierung davon abhängt, wie diese Informationen vor-

liegen. Eine einfache Möglichkeit besteht darin, daß die Zugriffskontrollinformationen zusammen mit den Daten (Anspruch 32), etwa im HTML-Header als Meta-Tags (Anspruch 33) oder als spezifische Felder in einem XML-Format (Anspruch 34) abgelegt sind.

3.2.6 Datenübertragung

[0112] Die Datenübertragung erfolgt im Beispiel ausschließlich über das Internet (vgl. Anspruch 2), also zunächst unsicher.

[0113] Alle Verbindungen werden über ssl, also verschlüsselt, realisiert (Ansprüche 16, ggf. 21, 26/27, 36). Dabei erfolgt immer eine wechselweise Authentifizierung (also sowohl Server gegenüber Client als auch Client gegenüber Server), um einen höchstmöglichen Schutz gegenüber Angreifern zu gewähren.

[0114] Die Abläufe sind so gestaltet, daß das einfache http-Protokoll mit Request-Response-Zyklen ausreicht. Somit kann an allen Verbindungen das standardisierte https-Protokoll (http über ssl) verwendet werden (Ansprüche 17, 27, 36).

[0115] Erweiterungen und Variationen der Netzwerkprotokolle sind an dieser Stelle zwar denkbar, versprechen aber keinen funktionalen Vorteil.

3.2.7 Trust-Center

[0116] Im beschriebenen Ablauf sind eine Reihe von asymmetrischen Schlüsselpaaren im Einsatz, und zwar mindestens:

- Schlüssel des Arztes auf der HPC (Anspruch 1)
- Schlüssel des Patienten (Anspruch 1) auf seiner biometrisch gesicherten (Ansprüche 9, 10) Karte
- ssl-Client-Schlüssel des Arzt-Arbeitsplatzes (Anspruch 16 und 35)
- ssl-Server-Schlüssel des Zugriffskontrollsystems (Anspruch 16 und 26)
- ssl-Server-Schlüssel des Freigabesystems (Anspruch 26 und 35)
- ssl-Client-Schlüssel des Zugriffskontrollsystems (nach Anspruch 24 und 26) und/oder des Freigabesystems (nach Anspruch 25 und 26)

[0117] Diese Schlüssel werden sinnvollerweise von einem Trust-Center als spezialisiertem Dienstleister generiert und/oder in ihrer Echtheit bestätigt (Anspruch 10). Inwieweit hierbei ein einziges oder mehrere Trust-Center verwendet werden, und ob diese organisatorisch an einer der dargestellten Stellen, z. B. der Zugriffskontrollstelle, angelagert sind, ist aus technischer Sicht nicht von Belang.

3.3 Ablauf aus Benutzersicht

3.3.1 Antragssoftware

[0118] Wenn der Arzt seine Behandlung aufnimmt, lädt oder aktiviert er eine geeignete "Antrags-Software" (z. B. ein im Browser lauffähiges Java-Applet; "Authorisierungs-Applet" im Bild), die auf seinem Rechner vorinstalliert ist oder die er aus einer vertrauenswürdigen Quelle (z. B. auch der Zugriffskontrollstelle) beziehen kann.

3.3.2 Freigabeantrag

[0119] Die Antragssoftware verlangt nun nach den Identitätsbeweisen, also etwa den Smart-Cards von Arzt und Patient. Die endgültige Zustimmung (nach Anspruch 1) erfordert noch die Freigabe des Schlüssels auf den Karten, z. B. durch ein Passwort bei der HPC und durch den Fingerabdruck bei der biometrischen Patientenkarte.

[0120] Nach Absenden des Freigabeantrags (nach Anspruch 3) an die Zugriffskontrollstelle meldet diese entweder einen Authentifzierungsfehler oder einen Hyperlink zum nächsten Schritt.

[0121] Der in Zeichnung 2 auf Seite 52 dargestellte Bildschirmabzug zeigt eine Pilotversion eines solchen Freigabeapplets, bei dem die Smart-Cards durch Binärsequenzen ersetzt sind, die über die Windows-Zwischenablage in entsprechende Fenster eingefügt werden ("paste doctor's/patient's key here").

[0122] Die Freigabe selbst erfolgt in der Pilotversion durch Passwörter, was im Bild beim Patienten bereits erfolgt ist ("open") und für den Arzt noch abgefragt wird ("please enter password")

3.3.3 Datenauswahl

[0123] Im Beispiel wird angenommen, daß die Zugriffskontrollstelle bereits über eine Liste von zum Patienten verfügbaren Daten verfügt. In diesem Fall kann an den Arbeitsplatz des Arztes eine entsprechende Liste als HMTL-Seite mit Hyperlinks versandt werden, wobei die Hyperlinks auf die jeweiligen Freigabeserver verweisen.

[0124] Arzt und/oder Patient wählen (ggf. gemeinsam) die abzurufenden Daten aus ("weitere Kontextinformationen" nach Anspruch 1).

3.3.4 Datenabruf

[0125] Im Idealfall verhält sich das nun geöffnete System gegenüber dem Arzt wie ein Satz von HTML-Seiten, er kann also im Bestand der Patientendaten mit seinem Web-Browser "surfen" (Anspruch 41 ff) und ggf. wichtige Informationen in sein eigenes Praxisdatensystem "downloaden".

Erweiterung

[0126] Bei geeigneter Datenstruktur können dabei auch Querverweise zwischen verschiedenen Datenbeständen auftreten. Ggf. wird das System nach einer erneuten Authentifizierung verlangen, wenn etwa der Bereich einer anderen Zugriffskontrollstelle betreten werden soll oder der Patient weitergehende Freigaberechte für besonders vertrauenswürdige Daten oder weitere Rechte (z. B. Schreibzugriff) erteilen soll.

3.4 Zusammenspiel der kryptographischen Komponenten

[0127] Die erforderlichen kryptographischen Abläufe sollen bei erfolgreicher Authentifizierung weitgehend im Hintergrund ablaufen. Lediglich bei einem Authentifizierungsfehler sollen Fehlermeldungen soweit aussagefähig sein, daß Bedienungsfehler behoben werden können. Allerdings dürfen Fehlermeldungen keine wertvollen Informationen an potentielle Angreifer offenbaren oder bereits vertrauenswürdige Daten (bzw. deren Existenz) preisgeben (falsch wäre z. B. "Sie sind nicht berechtigt, auf den Datenbestand der Sucht-Heil-Klinik zuzugreifen").

3.4.1 Schlüssel

[0128] Das Format der Schlüssel entspricht dem X.509-Standard, wie es von gängigen Internet-Browsern verstanden wird. (Beispiel: siehe Zeichnung 3)

[0129] Die Schlüssel für https müssen dabei direkt vom Browser verstanden werden können. Für die Smart-Card-Schlüssel ist dies nicht unbedingt notwendig, da sie über die Authentifizierungssoftware angesprochen werden. Die Verwendung standardisierter Formate ermöglicht aber den Zugriff auf vorhandene, getestete Softwarebibliotheken zur Erzeugung und Verarbeitung der kryptographischen Daten.

[0130] Ein Schlüsselpaar, das aus einem Browser im PKCS#12-Format ([PKCS12]) exportiert wird, stellt sich zunächst folgendermaßen dar:

```
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,FD4DC0A4919C7342

s+hFVpOmDxWgKbiTDMIimHKoJVerMWzalNeI2e1j8E/uRf89fNNRq0oTVhc+D7fd
ihPltydbSKnDUpgD5sMhQUyzUIdsD4dW1eqnF503RJ2t7e8rZk9kbAauQ02gcVRR
XpFrnxn2GDOFGHlyj9AymjyE/se7kyaZaIVodgQI7Y1/dGBKX/peJEwmtwkQMX1V
BG02JwyedknWcYM6t0hFAmRX9GL2bh/MhZEBQP80cUQhVBvV0oek9LIlW84Y+czp
QdwmrPzmNgScRr5cy6Hw/3T1eYvasdZufL38LwMpxwU3mOccTXgNr8w4wcGOMfA6
kVzbZMxunLPYQS5FekxLn8K7Ce/WKXEumTvLG1VaAd6mhE9vKKB1wk1VU1uIlzOl
F2LOur/Yg5gX8cLkC9L6cY7B8LSJ5z5C9HJG3F4553o=
-----END RSA PRIVATE KEY-----
Bag Attributes
friendlyName: Benjamin Blýmchen's IP-Web ID
localKeyID: 3C E3 30 30 57 B7 49 A9 C1 A8 AA C0 04 B3 48 CC EC F5 24 02
subject=/C=de/ST=ppp/L=Bibberberg/O=Interessenvereinigung sprechender
Elefanten/CN=Benjamin Bl\xFCmchen
issuer= /C=de/ST=demo/O=IP-Web/OU=demo/CN=IP-Web Demo Patients CA
-----BEGIN CERTIFICATE-----
MIICFDCCAX2gAwIBAgIEOrTuujANBgkqhkiG9w0BAQQFADBeMQswCQYDVQQGEwJk
ZTENMAsGA1UECBMEZGVtbzEPMA0GA1UEChMGSVAtV2ViMQ0wCwYDVQQLEwRkZW1v
MSAwHgYDVQQDExdJUC1XZWIgRGVtbyBQYXRpZW50cyBDQTAeFw0wMTAzMTgwMDAw
MDBaFw0wMzAzMTgyMzU5MDBaMIGCMQswCQYDVQQGEwJkZTEMMAoGA1UECBMDcHBw
MRMwEQYDVQQHEwpCaWJiZXJiZXJnMTQwMgYDVQQKEytJbnRlcmVzc2VudmVyZW1u
aWdlbmcgc3ByZWNoZW5kZXIgRWxlZmFudGVuMRowGAYDVQQDFBFCZW5qYW1pbiBC
bPxtY2hlbjBcMA0GCSqGSIb3DQEBAQUAA0sAMEgCQQDgnGfaZx6YNtrBn2islWL0
EpaNbSO7UBM+pBNUoGqnM/YVFelihLKBvaf8D0pMhI6iqpqfbsbTXPDP87yTsBdV
AgMBAAEwDQYJKoZIhvcNAQEEBQADgYEAwS+fUrinuGFzKd6E05fHqg2o4M+InnA6
zJX3a7RYefxijAWwQS9qCLQnMJ03aiCdspFL5WVH3QKbAXWHdJJsjOSTtj/5pUqa
+aGL1X0SGNE9H51f6DzXoFDbl1ZOjiU3Y5Dkbp7aJWbecYnt/jqrr7EGrC9O2TaU
241KFsP/nDg=
-----END CERTIFICATE-----
```

[0131] Die wesentlichen Komponenten sind der private Schlüssel ("BEGIN/END RSA PRIVATE KEY") sowie das X.509-Zertifikat mit dem öffentlichen Schlüssel ("BEGIN/END CERTIFICATE"). Beide Bestandteile sind in der Darstellung base-64-kodiert, was jedoch nur der Übertragbarkeit binärer Daten und nicht der Zugriffssicherheit dient. Die weiteren Angaben dienen nur der internen Organisation des PKCS-12-Formates.

[0132] Dieses Format entspricht im Wesentlichen auch der internen Darstellung auf einer Smart-Card.

[0133] Das Zertifikat, das die Echtheit des Schlüssels durch Trust-Center bestätigt, hat (nach base-64-Dekodierung) folgende Struktur (nach [X.509]).

```
Certificate:
    Data:
        Version: 3 (0x2)
        Serial Number: 984936122 (0x3ab4eeba)
        Signature Algorithm: md5WithRSAEncryption
        Issuer: C=de, ST=demo, O=IP-Web, OU=demo, CN=IP-Web Demo Patients CA
        Validity
            Not Before: Mar 18 00:00:00 2001 GMT
            Not After : Mar 18 23:59:00 2003 GMT
        Subject: C=de, ST=ppp, L=Bibberberg,
          O=Interessenvereinigung sprechender Elefanten,
          CN=Benjamin Bl\xFCmchen
        Subject Public Key Info:
            Public Key Algorithm: rsaEncryption
            RSA Public Key: (512 bit)
                Modulus (512 bit):
                    00:e0:9c:67:da:67:1e:98:36:da:c1:9f:68:ac:95:
                    62:f4:12:96:8d:6d:23:bb:50:13:3e:a4:13:54:a0:
                    6a:a7:33:f6:15:15:ed:62:84:b2:81:bd:a7:fc:0f:
                    4e:4c:84:8e:a2:aa:9a:9f:6e:c6:d3:5c:f0:cf:f3:
                    bc:93:b0:17:55
                Exponent: 65537 (0x10001)
    Signature Algorithm: md5WithRSAEncryption
        c1:2f:9f:52:b8:a7:b8:61:73:29:de:84:d3:97:c7:aa:0d:a8:
        e0:cf:88:9e:70:3a:cc:95:f7:6b:b4:58:79:fc:62:8c:05:b0:
        41:2f:6a:08:b4:27:30:9d:37:6a:20:9d:b2:91:4b:e5:65:47:
        dd:02:9b:01:75:87:74:92:6c:8c:e4:93:b6:3f:f9:a5:4a:9a:
        f9:a1:8b:d5:7d:12:18:d1:3d:1f:9d:5f:e8:3c:d7:a0:50:db:
        d7:56:4e:8e:25:37:63:90:e4:6e:9e:da:25:66:de:71:89:ed:
        fe:3a:ab:af:b1:06:ac:2f:4e:d9:36:94:db:8d:4a:16:c3:ff:
        9c:38
```

- "Subject" ist dabei der Name der zu identifizierenden Person
- "issuer" das Trust-Center, das die Echtheit bestätigt (vgl. Anspruch 10a).
- Validity (Gültigkeitsdatum) und Seriennummer sowie Version und Algorithmen sind weitere Organisationsbestandteile nach X.509
- Der öffentliche Schlüssel ("RSA public Key") ist wesentlicher Bestandteil des Zertifikats
- Die Signatur (eingeleitet mit der Angabe des Algorithmus), die die Echtheit des Schlüssels bestätigt, schließt das Zertifikat ab.

[0134] Der private Schlüssel ist in einem passwortgeschützten "Key Bag" untergebracht, d. h. zusätzlich zur base-64-Dekodierung ist noch eine Entschlüsselung mit dem Passwort erforderlich.

[0135] Erst nach Öffnen des "Key Bag" durch Passworteingabe (oder Fingerabdruck) eröffnet sich die interne Struktur des privaten RSA-Schlüssels, wie er zur Verwendung erforderlich ist:

```
Enter PEM pass phrase:
Private-Key: (512 bit)
modulus:
     00:e0:9c:67:da:67:1e:98:36:da:c1:9f:68:ac:95:
     62:f4:12:96:8d:6d:23:bb:50:13:3e:a4:13:54:a0:
     6a:a7:33:f6:15:15:ed:62:84:b2:81:bd:a7:fc:0f:
     4a:4c:84:8e:a2:aa:9a:9f:6e:c6:d3:5c:f0:cf:f3:
     bc:93:b0:17:55
publicExponent: 65537 (0x10001)
privateExponent:
     00:ae:e8:bd:6a:f3:60:7c:d2:42:ba:03:04:05:59:
     73:ac:73:89:2f:ea:ec:a7:62:2d:0a:5b:c4:fd:e2:
     ca:0b:17:53:1d:3e:d2:a4:3c:81:93:4d:89:82:37:
     23:28:1b:7c:9f:b6:8e:2a:eb:49:e3:b5:f1:c1:18:
     eb:e8:be:b4:21
prime1:
     C0:f5:0b:2c:25:3d:a3:95:1f:9b:ee:8b:83:b2:57:
     03:2e:d8:3c:85:cb:6c:50:00:1e:57:a7:1a:37:f4:
     0a:5d:c9
prime2:
     00:ea:a7:5b:1c:d3:a0:73:cc:3a:63:ca:33:73:e2:
     f3:cd:e6:4b:3d:5b:3c:f6:b1:37:37:98:4e:34:3d:
     c2:43:2d
exponent1:
     00:d2:3b:c2:a5:34:cb:fa:ee:02:97:57:a5:26:c5:
     d6:5a:43:75:31:89:04:a5:62:64:a5:e9:1c:ea:72:
     7a:ce:59
exponent2:
     2f:06:a7:1d:d9:d3:98:21:5f:ba:4b:f5:8f:cd:f5:
     ea:57:b2:d0:73:0e:7e:a9:f9:54:ec:f3:0f:49:29:
     3b:69
coefficient:
     49:09:37:c9:09:26:23:d4:1e:9c:58:b4:eb:11:38:
     a6:41:32:fe:21:a7:6e:07:21:c9:4b:34:57:11:43:
     74:60
```

[0136] Für Schlüssel, die auf Smart-Cards abgelegt sind, ist diese Struktur des privaten Schlüssels von außen nicht zugänglich (auch nicht nach Freigabe). Alle zu verschlüsselnden Daten müssen in die Smart-Card übertragen werden, werden dort (nach Freigabe) verschlüsselt und wieder aus der Smart-Card ausgelesen.

3.4.2 Authentifizierungs-Applet

[0137] Das Authentifizierungs-Applet übernimmt folgende Funktionen:

- Erstellen eines zufällig generierten Session Keys (Anspruch 4 und 11)
- Ermitteln der aktuellen Systemzeit (für Anspruch 4)
- Ermitteln der Identität von Arzt und Patient (Anspruch 1 – z. B. "Distinguished Names" nach X500 oder spezielle Arzt-/Patientenindices)
- Abfragen der Zustimmung von Arzt und Patient (Anspruch 1)
- bei Erteilung der Zustimmung: Verschlüsselung des Session Keys mit dem jeweiligen privaten Schlüssel von Arzt und Patient (Anspruch 5 und 6 sowie 12)
- Zusammenfügen der Komponenten zu einem Session-Request (nach Anspruch 3 ff)
- Signieren des Session-Request mit dem (unverschlüsselten) Session-Key (Anspruch 13)

[0138] Erweiterung:
Wenn der https-Clienf-Schlüssel im Session-Request übertragen werden soll, so muß das Applet diesen vom Browser abrufen (Anspruch 14 ff, insbesondere 17)

[0139] Erweiterung:
Gegebenenfalls werden noch weitere Informationen zur Spezifikation des Kontextes, wie in Anspruch 1 angeführt, angegeben

[0140] Erweiterung:
Das X.509-Zertifikat einer an der Authentifizierung beteiligten Person (oder mehrerer Personen), ggf. auch zugehörige Zertifizierungsketten, können ebenfalls im Antrag mit aufgenommen werden (Anspruch 20). In diesem Fall braucht beim Zugriffskontrollsystem nur das entsprechende Wurzelzertifikat vorgehalten werden (vgl. Anspruch 19). Damit wird die Verantwortung, die Berechtigung einer Person, das System generell zu benutzen, zu prüfen, von der Zugriffsprüfstelle an das Trust-Center abgegeben.

3.4.3 Session-Request

[0141] Der erstellte Session-Request (s. Anspruch 3 ff) hat im Beispiel folgende Struktur (ohne X.509-Zertifikate nach Anspruch 20):

```
x-+-0-+-0.0-+- 0.0.0 -- Session ID 254 bit random
  |   |     +- 0.0.1 -- Session time Stamp
  |   |
  |   +-0.1---------- encrypted Session key (DESede x docPK x patPK)
  |   |
  |   |               Doctors Data:
  |   +-0.2-+- 0.2.0 -- Subject DN
  |   |     +- 0.2.1 -- Issuer DN
  |   |     +- 0.2.2 -- Serial Number
  |   |     +- 0.2.3 -- Fingerprint
  |   |
  |   |               Patients Data:
  |   +-0.3-+- 0.3.0 -- Subject DN
  |   |     +- 0.3.1 -- Issuer DN
  |   |     +- 0.3.2 -- Serial Number
  |   |     +- 0.3.3 -- Fingerprint
  |   |
  |   +-0.4-+- 0.4.0 -- SessionIDblock signed(docPK)
  |         +- 0.4.1 -- SessionIDblock signed(patPK)
  |
  |
  +-1---- Bag Seal: MD5 of 0 encrypted with session Key
```

[0142] Diese Struktur kann z. B. in ASN.1 (siehe [X.680]) dargestellt werden, in Base 64 kodiert und dann als Query an eine URL angefügt und so via http(s) an das Zugriffskontrollsystem übertragen werden.

3.4.4 Authentitätsprüfung durch das Zugriffskontrollsystem

[0143] Das Zugriffskontrollsystem dekodiert die query:

raw DER-Structure as Hex-Dump:

```
          00 01 02 03 04 05 06 07 - 08 09 0A 0B 0C 0D 0E 0F  0123456789ABCDEF

00000000  30 82 02 67 30 82 02 51 - 30 37 04 20 17 82 69 AD  0..g0..Q07. ..i.
00000010  30 97 3C 14 20 F2 0B 12 - AC 7D BC 4D 31 68 B3 34  0.<. ....}.M1h.4
00000020  70 E3 5F AC D4 35 BA 67 - B4 CF 4E 0F 18 13 32 30  p._..5.g..N...20
00000030  30 31 30 34 30 32 31 31 - 35 39 35 33 2B 30 32 30  010402115953+020
00000040  30 04 40 02 F2 43 E7 9F - AE 7D E5 3B 4C DF 73 E5  0.@..C...}.;L.s.
00000050  81 BB A0 69 6E DC 5D 0B - EE 55 43 2E 57 24 03 2C  ...in.]..UC.W$.,
00000060  3D 48 34 BA 25 EB 2D F3 - 6E C9 A1 25 B5 91 5F CB  =H4.%.-.n..%.._.
00000070  97 FA 25 BB 5D 41 5F 44 - EE 06 18 58 75 2A 96 5C  ..%.]A_D...Xu*.\
00000080  AB AD 39 30 81 9A 13 47 - 43 4E 3D 4A 6F 68 61 6E  ..90...GCN=Johan
00000090  6E 20 4A 6F 68 6E 73 65 - 64 65 72 2C 4F 3D 47 65  n Johnseder,O=Ge
000000A0  6D 65 69 6E 73 63 68 61 - 66 74 73 70 72 61 78 69  meinschaftspraxi
000000B0  73 20 4A 56 58 2C 4C 3D - 41 6E 64 65 72 68 61 75  s JVX,L=Anderhau
000000C0  73 65 6E 2C 53 54 3D 62 - 62 62 2C 43 3D 64 65 13  sen,ST=bbb,C=de.
000000D0  37 43 4E 3D 49 50 2D 57 - 65 62 20 44 65 6D 6F 20  7CN=IP-Web Demo
000000EC  44 6F 63 74 6F 72 73 20 - 43 41 2C 4F 55 3D 64 65  Doctors CA,OU=de
000000F0  6D 6F 2C 4F 3D 49 50 2D - 57 65 62 2C 53 54 3D 64  mo,O=IP-Web,ST=d
00000100  65 6D 6F 2C 43 3D 64 65 - 02 04 3A B4 EC 35 04 10  emo,C=de..:..5..
00000110  5E 9A 5F 89 C9 C0 AE 01 - 2E 4A 77 A7 B2 02 5D 33  ^._.....Jw...]3
00000120  30 81 AF 13 5B 43 4E 3D - 42 65 6E 6A 61 6D 69 6E  0...[CN=Benjamin
00000130  20 42 6C 3F 6D 63 68 65 - 6E 2C 4F 3D 49 6E 74 65   Bl?mchen,O=Inte
00000140  72 65 73 73 65 6E 76 65 - 72 65 69 6E 69 67 75 6E  ressenvereinigun
00000150  67 20 73 70 72 65 63 68 - 65 6E 64 65 72 20 45 6C  g sprechender El
00000160  65 66 61 6E 74 65 6E 2C - 4C 3D 42 69 62 62 65 72  efanten,L=Bibber
00000170  62 65 72 67 2C 53 54 3D - 70 70 70 2C 43 3D 64 65  berg,ST=ppp,C=de
00000180  13 38 43 4E 3D 49 50 2D - 57 65 62 20 44 65 6D 6F  .8CN=IP-Web Demo
00000190  2C 50 61 74 69 65 6E 74 - 73 20 43 41 2C 4F 55 3D  ,Patients CA,OU=
000001A0  64 65 6D 6F 2C 4F 3D 49 - 50 2D 57 65 62 2C 53 54  demo,O=IP-Web,ST
000001B0  3D 64 65 6D 6F 2C 43 3D - 64 65 02 04 3A B4 EE BA  =demo,C=de..:...
000001C0  04 10 64 57 A6 FA 06 8C - 0E A9 01 E3 D1 3C 1B 56  ..dW.........<.V
000001D0  E5 02 30 81 84 04 40 06 - A7 25 15 ED 96 04 FC 60  ..0...@..%.....`
000001E0  32 12 28 3C 0A 4F E8 EE - 02 24 AF 2B 7E B3 1B B5  2.(<.O...$.+~...
000001F0  A6 89 3F 44 98 75 C1 EF - EE 66 82 FC 2B 16 5F D2  ..?D.u...f..+._.
00000200  5F F4 85 5E 7B F8 0D A3 - E6 74 D3 5C 3B 9A 8D 6A  _..^{....t.\;..j
00000210  A4 4C 6D 07 6C 55 BA 04 - 40 6C 12 FA E8 4C 55 40  .Lm.lU..@l...LU@
00000220  A4 70 5D 04 6F 7E AD 9C - 77 F7 08 98 A2 3E E0 C7  .p].o~..w....>..
00000230  31 91 97 D1 39 94 C7 B6 - 08 AC 30 7A 17 15 68 B9  1...9.....0z..h.
00000240  97 94 6A 2C 51 E3 59 C5 - 4C 82 4E 21 FE A7 2A 51  ..j,Q.Y.L.N!..*Q
00000250  BF 26 EE AD 87 4D 77 8A - 6B 04 10 1F BC 0B 1D B9  .&...Mw.k.......
00000260  B8 36 50 48 41 BA 9D F9 - BD 55 83                 .6PHA....U.
```

[0144] Daraus ergibt sich die ASN.1-Struktur, die die Komponenten des Session-Requests offenlegt (vgl. ASN.1-Struktur wie oben angegeben):

```
ASN1-Structure of Request:

  0:d=0  hl=4 l= 615 cons: SEQUENCE
  4:d=1  hl=4 l= 593 cons:  SEQUENCE
  8:d=2  hl=2 l=  55 cons:   SEQUENCE
 10:d=3  hl=2 l=  32 prim:    OCTET STRING
 44:d=3  hl=2 l=  19 prim:    GENERALIZEDTIME    :20010402115953+0200
 65:d=2  hl=2 l=  64 prim:   OCTET STRING
131:d=2  hl=3 l= 154 cons:   SEQUENCE
134:d=3  hl=2 l=  71 prim:    PRINTABLESTRING    :
      CN=Jchann Johnseder,
      O=Gemeinschaftspraxis JVX,L=Anderhausen,ST=bbb,C=de
207:d=3  hl=2 l=  55 prim:    PRINTABLESTRING    :
      CN=IP-Web Demo Doctors CA,OU=demo,O=IP-Web,ST=demo,C=de
264:d=3  hl=2 l=   4 prim:    INTEGER            :3AB4EC35
270:d=3  hl=2 l=  16 prim:    OCTET STRING
288:d=2  hl=3 l= 175 cons:   SEQUENCE
291:d=3  hl=2 l=  91 prim:    PRINTABLESTRING    :
      CN=Benjamin Bl?mchen,
      O=Interessenvereinigung sprechender Elefanten,
      L=Bibberberg,ST=ppp,C=de
384:d=3  hl=2 l=  56 prim:    PRINTABLESTRING    :
      CN=IP-Web Demo Patients CA,OU=demo,O=IP-Web,ST=demo,C=de
442:d=3  hl=2 l=   4 prim:    INTEGER            :3AB4EEBA
448:d=3  hl=2 l=  16 prim:    OCTET STRING
466:d=2  hl=3 l= 132 cons:   SEQUENCE
469:d=3  hl=2 l=  64 prim:    OCTET STRING
535:d=3  hl=2 l=  64 prim:    OCTET STRING
601:d=1  hl=2 l=  16 prim:  OCTET STRING
```

[0145] Es liegen nun die Informationen vor, um die Überprüfung der Authentizität vorzunehmen:

- aus den Identitätsinformationen von Arzt und Patient sowie deren Zertifizierern werden die entsprechenden Zertifikate ermittelt
- Über Seriennummer und Fingerprint der Zertifikate wird ein erster Echtheitstest vorgenommen (Anspruch 18)
- Die Zertifikate enthalten den jeweiligen öffentlichen Schlüssel
- der verschlüsselt vorliegende Session Key wird mit den beiden öffentlichen Schlüsseln von Arzt und Patient entschlüsselt (Anspruch 12):
  Dann und nur dann wenn sowohl Arzt als auch Patient mit dem richtigen privaten Schlüssel gearbeitet haben, läßt sich so der ursprüngliche Session Key wiederherstellen
- Es wird der Hash-Wert des Session-Requests gebildet und mit dem wiederhergestellten Session Key verschlüsselt. Wenn das Ergebnis mit dem übermittelten Wert ("Bag Seal") identisch ist, dient das als Beweis für eine korrekte Authentifizierung (Anspruch 13)

[0146] Der so wiederhergestellte Session Key kann bei Bedarf weiter verwendet werden, um die weitere Kommunikation zu verschlüsseln, so daß nur die berechtigten Parteien daran teilnehmen können (Anspruch 15).

[0147] Erweiterung:
Das Zugriffskontrollsystem kann den ssl-Schlüssel des Clients aus der Netwerksoftware abrufen, wenn diese Information nicht Bestandteil des Session-Requests ist. Damit entfällt clientseitig die Notwendigkeit, daß das Applet auf den https-Schlüssel zugreifen muß. Dies wiederum vereinfacht die Verwendung von Standardbrowsern auf Clientseite.

3.4.5 Freigabeserver

[0148] Der entschlüsselte Session-Key, die ssl-Identität des PC in der Arztpraxis und die anderen Informationen des Session-Requests werden an den Freigabeserver (nach Anspruch 23) übermittelt. Hierzu kann z. B. das Zugriffskontrollsystem eine https-Verbindung zu diesem aufbauen (vgl. Anspruch 24, 26, 27). Der Freigabeserver speichert die Informationen, um sie für den Fall eines Datenabrufs auswerten zu können.

[0149] Daraufhin übermittelt das Zugriffskontrollsystem eine Erfolgsmeldung an den Browser beim Arzt, die sinnvollerweise einen Hyperlink auf den Freigabeserver (bzw. je einen Link auf jeden Server, wenn Datenbestände an verschiedenen Stellen vorliegen) enthält. Außerdem können die Links (gemäß Anspruch 28 und 29) Query-Variablen enthalten, die zur weiteren Spezifikation der verlangten Daten dienen (z. B. Namen des Patienten, um Verwechslungen bei kurz aufeinanderfolgenden Abfragen zu vermeiden).

[0150] Durch Aktivieren der Links im Browser (Anspruch 41 ff) sendet das Arztsystem nun einen http(s)-Request an den Freigabeserver. Dieser ruft die gewünschten Daten aus der Datenbank ab und erhält von dieser (ggf. auch vorher) die für den Zugriff erforderlichen Voraussetzungen (Anspruch 30 ff). Diese Voraussetzungen vergleicht er mit den gespeicherten Daten aus dem vorher übermittelten Session-Request (Anspruch 30).

[0151] Des weiteren ermittelt der Freigabeserver die ssl-Identität des Arzt-PC und vergleicht sie mit der vom Zugriffskontrollsystem übermittelten (Anspruch 35, 36). Damit wird verhindert, daß ein unbefugter Benutzer mit einer aufge-

zeichneten Zugriffssequenz, die er zwar nicht analysieren, aber 1 : 1 wiedergeben kann, Datenzugriff erhält ("replay-attack").

[0152] Wenn alle Prüfungen erfolgreich sind, überträgt der Freigabeserver die erbetenen Daten als http(s)-Response an den Arzt-PC. Dieser Response kann weitere Hyperlinks zu anderen Daten auf dem gleichen oder einem anderen System enthalten (Anspruch 41 ff). Sofern für einen Abruf bestimmte Inhalte einer URL-query erforderlich sind, werden diese im Response weitergereicht ("state-Management" im http-Protokoll, vgl. auch Anspruch 28 und 29).

3.5 Ergänzungen und Erweiterungen

[0153] Wichtige Variationen der Erfindung, die hier im Ausführungsbeispiel nicht oder nur andeutungsweise erfaßt sind, werden im Folgenden noch angeführt.

3.5.1 Komplexe Verschlüsselung in der Datenübertragung

[0154] Im beschriebenen Beispiel erfolgt die Übertragung zwischen freigebender Stelle über https, also verschlüsselt mit dem ssl-Client-Schlüssel des Arzt-PC. Dieser Schlüssel ist üblicherweise auf dem Rechner in einer Datei abgelegt und ist damit unbefugten Zugriffen z. B. bei der Rechneradministration ausgesetzt ("man-at-the-end-attack"). Dieser Schlüssel bietet also bei weitem nicht die Sicherheit wie die Smart-Card-basierten Schlüssel zur persönlichen Authentifizierung.

[0155] Alternative Schlüssel, die diesen Mangel beheben können, sind in Anspruch 37/38 beschrieben. Insbesondere der Session Key käme hierfür in Frage. Die meisten dieser Lösungen haben jedoch den Nachteil, daß clientseitig keine Standardbrowser mehr verwendet werden können.

3.5.2 Schreibender Datenzugriff

[0156] Die bisherigen Beschreibungen des Ausführungsbeispiels beziehen sich teilweise implizit auf lesenden Zugriff.

[0157] Die Abläufe für schreibenden Zugriff (Daten anlegen, ändern, löschen, Rechte ändern) sind weitgehend identisch. Lediglich der Freigabeserver und die Kommunikation mit der Datenbank muß entsprechend angepaßt werden. Als Kommunikationsprotokoll mit dem Client kann weiterhin http(s) verwendet werden, das durch Formulare und POST-Requests eine Dateneingabe ermöglicht.

3.5.3 Verzeichnisdienste für Zertifikate

[0158] Das beschriebene Ausführungsbeispiel geht nicht darauf ein, woher die Zertifikate kommen, die für die Authentitätsüberprüfung eingesetzt werden. Neben der Übermittlung mit dem Session Request nach Anspruch 20 sind die Variationen in Anspruch 21 und 22 dargelegt.

[0159] Eine sinnvolle Möglichkeit wäre etwa, wenn die Ausgabe der Smart-Cards von einem Trust-Center erfolgt, das nicht nur die Identität des Inhabers, sondern auch weitergehende Voraussetzungen zur Teilnahme am System (z. B. bei einem Arzt die fachliche Zulassung) prüft und festhält. Diese Informationen können dann – zusammen mit dem Zertifikat – von diesem Trust-Center über LDAP (ggf. über eine gesicherte Verbindung) dem Zugriffskontrollsystem übermittelt werden.

[0160] Bei einem Ausbau des Systems können mehrere unabhängige Stellen diese Zertifizierung übernehmen (z. B. Krankenkassenfilialen für Patienten). In diesem Fall empfiehlt sich ein mehrstufiges Zertifzierungsverfahren, wie es in [X.509] vorgesehen und in Anspruch 18 ff angeführt ist.

3.5.4 Verzeichnis der verfügbaren Daten

[0161] Die beschriebene Realisierung geht davon aus, daß letztlich erst das datenhaltende System detaillierte Informationen besitzt, welche Daten zu einem spezifischen Kontext zur Verfügung stehen.

[0162] Diese Information kann aber auch anders über die einzelnen Systemkomponenten (Zugriffskontrolle, Freigabe, mehrere datenhaltende Stellen, dedizierte Verzeichnisse) verteilt werden. Auch eine Kombination mit anderen Datenbeständen (Verzeichnisdienst nach vorgehendem Abschnitt, Zugriffskontrolliste) ist denkbar.

[0163] Die detaillierten Variationsmöglichkeiten des Datenbankdesigns ergeben sich aus praktischen und sicherheitspolitischen Erwägungen und gehen über den in diesem Patent beanspruchbaren Rahmen hinaus.

3.5.5 Iterative Zustimmung zur Datenfreigabe

[0164] In der Zeichnung zum Gesamtsystem ist ein "Selection Key" angedeutet, der verwendet wird, wenn der Patient explizite Zustimmung zur Offenlegung einer beschränkten Teilauswahl aus allen verfügbaren Informationen erteilen soll. In den weiteren Erläuterungen wurde aus Gründen der Übersichtlichkeit hierauf nicht weiter eingegangen.

[0165] Hierbei handelt es sich um kontextspezifische Informationen (nach Anspruch 1), die durch mehrere Anforderungszyklen (nach Anspruch 41) ermittelt werden.

[0166] Der Selection Key übernimmt dabei die Rolle des Session Key; es wird gleichsam durch Iteration nach Anspruch 41 ein neuer Kontext nach Anspruch 1 eröffnet.

3.5.6 Vertretung von Personen

[0167] Anspruch 45 erwähnt verschiedene Vertretungskonstellationen. Beispiele hierfür können sein:

- nicht ein Arzt selber, sondern ein Assistent ruft Daten ab
- Patienten werden von Erziehungsberechtigten oder einem Vormund vertreten
- Ein "Notzugang" ermöglicht den Zugriff auf Daten von Patienten ohne Bewußtsein
- Zugriffsrechte werden nicht einer Person, sondern einer Organisation gewährt
- Zugriffsschlüssel können in einer Software, z. B. der Praxissoftware des Arztes, hinterlegt sein
- Applikationen in anderem als dem hier beispielhaft dargestellten Kontext können andere Vertretungsmöglichkeiten beinhalten

[0168] Welche Vertretungsmöglichkeiten zugelassen sind ist eine Frage der Sicherheitspolitik.

3.5.7 Festlegung einer Sicherheitspolitik

[0169] Viele Entscheidungen über die Implementierung können nicht nach technischen Erwägungen auf der Suche nach einer "bestmöglichen Sicherheit" getroffen werden, sondern entstehen durch Abwägung zwischen Sicherheitstechnik, rechtlichen Anforderungen, Kompatibilität mit bestehenden Systemen oder Bedienbarkeit. Selbst subjektive Vertrauens-"Gefühle" müssen hier mit einbezogen werden. Diese Erwägungen entziehen sich naturgemäß der technischen Natur eines Patentes, weshalb in dieser Beschreibung derartige Entscheidungen und Varianten regelmäßig offen gelassen werden.

[0170] Aufbauend auf dem hier beschriebenen Grundgerüst können Anpassungen an eine vorgegebene Sicherheitspolitik – bei gegebenen Anforderungen – durch einen geübten Fachmann vorgenommen werden.

Patentansprüche

1. Verfahren zur Zugriffsteuerung auf elektronisch gespeicherte Daten in verteilten heterogenen Umgebungen, **gekennzeichnet dadurch**, daß
die Freigabe des Datenzugriffs durch mehrere Personen erfolgt
eine dritte Stelle zur Freigabeprüfung zwischen Datenabrufstelle und Datenspeicherstelle eingeschaltet ist
für die Überprüfung der Authentizität der Freigabe erteilenden Personen kryptographische Verfahren mit öffentlichen und privaten Schlüsseln verwendet werden
die Kontextsituation der Personen im Rahmen der Datenanforderung mit in den Umfang der erteilten Freigabe einfließen kann.

2. Verfahren nach Anspruch 1, gekennzeichnet dadurch, daß
für die Datenübertragung geeignete Netzwerkprotokolle verwendet werden können,
für die Datenübertragung die Internet-Protokoll-Familie (insbesondere IP, TCP, UDP) verwendet werden können,
für die Datenübertragung das Internet verwendet werden kann.

3. Verfahren nach Anspruch 1, gekennzeichnet dadurch, daß für jeden Vorgang, bei dem eine Freigabe angefordert wird
ein spezielles digitales Freigabeobjekt ("Authentifizierungsantrag") zur Kommunikation der beteiligten Systeme untereinander erstellt wird
das Freigabeobjekt ein eindeutiges Merkmal des Authentifizierungsvorganges enthält
das Freigabeobjekt weitere Daten über den Authentifizierungskontext enthalten kann

4. Verfahren nach Anspruch 3, gekennzeichnet dadurch, daß
das Freigabeobjekt einen zufällig generierten oder anderweitig als eindeutig geltenden digitalen Schlüssel zur Identifikation des Authentifizierungsvorgangs enthalten kann
das Freigabeobjekt einen Zeitstempel des Authentifizierungsantrages enthalten kann
das Freigabeobjekt Angaben über den zeitlichen Geltungsbereich der Authentifizierung enthalten kann

5. Verfahren nach Anspruch 1, gekennzeichnet dadurch, daß die am Authentifizierungskontext beteiligten Personen durch asymmetrische digitale Schlüsselpaare identifiziert sind

6. Verfahren nach Anspruch 5, gekennzeichnet dadurch, daß eines der folgenden Verfahren für die asymmetrischen digitalen Schlüssel verwendet wird:
RSA
DSA
elliptische Funktionen

7. Verfahren nach Anspruch 5, gekennzeichnet dadurch, daß der private Schlüssel einer an der Authentifizierung beteiligten Person auf einer physikalisch eigenständigen Einheit gehalten wird, die
von der Person mitgeführt werden kann
vom System, das an der Datenabrufstelle zur Anforderung der Datenfreigabe verwendet wird, getrennt werden kann
mit unterschiedlichen Systemen zur Anforderung der Datenfreigabe verwendet werden kann
die erforderlichen Verschlüsselungsoperationen selbst vornehmen kann, so daß der private Schlüssel der Person nicht an das System zur Anforderung der Datenfreigabe übertragen werden muß

8. Verfahren nach Anspruch 7, gekennzeichnet dadurch, daß diese eigenständige Einheit eine handelsübliche Smart-Card (integrierter Schaltkreis, eingebaut in ein Kunststoffgehäuse im Scheckkartenformat) mit Kryptoprozessor ist

9. Verfahren nach Anspruch 5 bis 8, gekennzeichnet dadurch, daß der private Schlüssel durch ein biometrisches Merkmal vor unbefugter Verwendung geschützt ist .

10. Verfahren nach Anspruch 9, gekennzeichnet dadurch, daß als biometrisches Merkmal eines oder mehrere der Folgenden verwendet werden können:
individueller Fingerabdruck

individuelle Iris-Musterung
individuelle Gesichtszüge
individuelle Sprache
genetische Muster
Muster aus der Gen-Exprimierung, insbesondere RSA und Proteine

11. Verfahren nach Anspruch 1 bis 10, gekennzeichnet dadurch, daß für die Identifikation des Authentifizierungskontextes eine eindeutige binäre Sequenz als "Session Key" verwendet wird

12. Verfahren nach Anspruch 11, gekennzeichnet dadurch, daß der Session Key mit privaten Schlüsseln von am Kontext beteiligten Personen verschlüsselt wird

13. Verfahren nach Anspruch 3, 11 und 121 gekennzeichnet dadurch, daß die Integrität des Authentifizierungsantrages dadurch gewährleistet wird, daß der gesamte Antrag oder Teile davon mit dem Session Key oder einer davon abgeleiteten binären Sequenz digital signiert wird

14. Verfahren nach allen bisher genannten Ansprüchen, gekennzeichnet dadurch, daß das System, das zur Anforderung der Datenfreigabe verwendet wird, Bestandteil des Authentifizierungskontextes ist

15. Verfahren nach Anspruch 14, gekennzeichnet dadurch, daß
das anfordernde System gegenüber dem prüfenden System durch ein unabhängiges asymmetrisches Schlüsselpaar identifiziert werden kann
die Kommunikation zwischen anforderndem System und prüfendem System verschlüsselt erfolgen kann

16. Verfahren nach Anspruch 15, gekennzeichnet dadurch, daß die Kommunikation zwischen anforderndem System und prüfendem System über eine Variante des Protokolls ssl oder tls ("secure Socket Layer") erfolgen kann

17. Verfahren nach Anspruch 16, gekennzeichnet dadurch, daß zur Authentifizierung des Gerätes gegenüber der prüfenden Stelle und zur Verschlüsselung der Datenübertragung das Protokoll https ("http over ssl") verwendet wird

18. Verfahren nach den bisher genannten Ansprüchen, gekennzeichnet dadurch, daß die Überprüfung der Gültigkeit von Authentizitätsbehauptungen anhand einer dieser Möglichkeiten erfolgt:
eines bei der prüfenden Stelle hinterlegten öffentlichen Schlüssels als Gegenstück zum privaten Schlüssel des zu authentifizierenden Akteurs
der elektronischen Bescheinigung einer vertrauenswürdigen Instanz ("Trust Center"), deren öffentlicher Schlüssel bei der prüfenden Stelle vorliegt
der elektronischen Bescheinigung einer vertrauenswürdigen Instanz ("Trust Center"), deren Authentizität von einer Kette aus Authentizitätsbescheinigungen besteht, wobei am Anfang der Kette eine Instanz steht, deren öffentlicher Schlüssel bei der prüfenden Stelle vorliegt

19. Verfahren nach Anspruch 18, gekennzeichnet dadurch daß für Authentizitätsprüfung die Verfahren verwendet werden, wie sie in ISO X.509 festgelegt sind

20. Verfahren nach Anspruch 18 und 19, gekennzeichnet dadurch, daß zur Authentifikation erforderliche Zusatzinformationen (z. B. X.509-Zertifikate, Zertifikatsketten oder vergleichbare Daten) zusammen mit dem Freigabeantrag übermittelt werden können

21. Verfahren nach Anspruch 18, 19 und 20, gekennzeichnet dadurch, daß die zur Authentizitätsprüfung erforderlichen öffentlichen Schlüssel sich in einer Datenbank an einer der folgenden Standorte befinden:
der prüfenden Stelle selbst
innerhalb der unmittelbaren, kontrollierbaren Sicherheitssphäre der prüfenden Stelle
einer öffentlich verfügbaren Stelle, zu der von der prüfenden Stelle aus eine unverfälschbare Datenverbindung besteht

22. Verfahren nach Anspruch 21, gekennzeichnet dadurch, daß die Übertragung zwischen der prüfenden Stelle und der Datenbank durch eines der folgenden standardisierten Protokolle erfolgt
ITU-T- Empfehlung X500 ff.
Lightwighted Directory Access Protocol (LDAP)

23. Verfahren nach einem der bisher genannten Ansprüchen, gekennzeichnet dadurch, daß
die Prüfung der Authentizität der Freigabe erteilenden Akteure durch eine andere Stelle erfolgen kann als der, die die Daten frei gibt
die Freigabe der Daten durch eine andere Stelle erfolgen kann als der, die die Daten gespeichert hat

24. Verfahren nach 23, gekennzeichnet dadurch, daß
die prüfende Stelle an die freigebende Stelle einen geeigneten Datensatz überträgt, der der freigebenden Stelle die Evaluation des Kontextes ermöglicht ("push Session Request")
dieser Datensatz an der freigebenden Stelle gespeichert werden kann, bis eine Anforderung von der abrufenden Stelle erfolgt
und/oder aufgrund des Freigabedatensatzes bereits Daten an die abrufende Stelle gesandt werden können

25. Verfahren nach 23, gekennzeichnet dadurch, daß die freigebende Stelle von der prüfenden Stelle einen geeigneten Datensatz abruft, der der freigebenden Stelle die Evaluation des Kontextes ermöglicht, sobald eine Anforderung zur Datenfreigabe an sie gerichtet wird ("pull Session Request")

26. Verfahren nach Anspruch 23 bis 25, gekennzeichnet dadurch, daß
die Kommunikation zwischen freigebender und prüfender Stelle mit wechselseitiger kryptographischer Authentifizierung erfolgen kann
die Kommunikation zwischen freigebender und datenhaltender Stelle mit wechselseitiger kryptographischer Authentifizierung erfolgen kann
die Kommunikation zwischen freigebender und prüfender Stelle verschlüsselt erfolgen kann
die Kommunikation zwischen freigebender und datenhaltender Stelle verschlüsselt erfolgen kann

27. Verfahren nach Anspruch 26, gekennzeichnet dadurch, daß
die Kommunikation der benannten Systeme über ssl (oder tls) erfolgt

die Kommunikation der benannten Systeme über https erfolgen kann

28. Verfahren nach Anspruch 23, gekennzeichnet dadurch, daß anstatt oder ergänzend zu einer direkten Kommunikation zwischen freigebender und prüfender Stelle die prüfende Stelle ein geeignetes, von der prüfenden Stelle signiertes, kryptographisches Datum ("ticket" oder "token") an die anfordernde Stelle übermittelt, das die Richtigkeit der Authentifizierung beweist und mit dem sich dann die anfordernde Stelle der freigebenden Stelle gegenüber ausweisen kann

29. Verfahren nach Anspruch 28, gekennzeichnet dadurch, daß dieses Token über geeignete http-state-Management-Technologien übertragen wird, insbesondere
im Browser der anfordernden Stelle gespeicherte Variableninhalte ("Cookies")
von Requestzyklus zu Requestzyklus weitergereichte URL-Query- bzw. HTTP-POST-Variablen-Inhalte

30. Verfahren nach bisher genannten Ansprüchen, gekennzeichnet dadurch, daß
die freigebende Stelle über eine "Zugriffskontrolliste" verfügt, die angibt, wer unter welchen Voraussetzungen Zugriff mit welcher Berechtigung auf welche Daten hat
der Inhalt des Zugriffsantrages, insbesondere auch die entsprechenden Informationen über den Zugriffskontext, mit den Vorgaben aus der Zugriffskontrolliste verglichen werden und daraus die zugestandenen Zugriffsrechte abgeleitet werden

31. Verfahren nach Anspruch 30, gekennzeichnet dadurch, daß die freigebende Stelle die Zugriffskontrollinformationen bei Bedarf von dem datenhaltenden System abruft

32. Verfahren nach Anspruch 30, gekennzeichnet dadurch, daß die datenhaltende Stelle die Zugriffskontrollinformationen direkt mit den erbetenen Daten – noch vor der Berechtigungsprüfung – an die freigebende Stelle übermittelt

33. Verfahren nach Anspruch 32, gekennzeichnet dadurch, daß
das datenhaltende System die erbetene Antwort an das freigebende System im HTML-Format übermittelt
die Zugriffskontrollinformationen als spezifische, im Browser nicht störende oder vom Freigabesystem zu entfernende HTML-Tags übermittelt werden

34. Verfahren nach Anspruch 32, gekennzeichnet dadurch, daß
das datenhaltende System die erbetene Antwort an das freigebende System im XML-Format übermittelt
die Zugriffskontrollinformationen als eigens definierte XML-Elemente oder -Attribute übermittelt

35. Verfahren nach bisher genannten Ansprüchen (insbesondere 14 bis 17), gekennzeichnet dadurch, daß
der Zugriff auf freigegebene Daten nur von dem System aus erfolgen kann, das die Freigabeanforderung erstellt hat
die Authentifizierungsinformationen des Zugriff verlangenden Systems vom Authentiztät prüfenden System an das freigebende System mit übermittelt werden
diese Authentifizierungsinformationen netzwerkspezifische Daten (z. B. IP-Adresse, DNS-Namen) enthalten können
diese Authentifizierungsinformationen Bestandteile oder eindeutige Merkmale eines kryptographischen Schlüsselpaares enthalten können

36. Verfahren nach Anspruch 35, gekennzeichnet dadurch daß
die prüfende Stelle an die freigebende Stelle den ssl-Client-Schlüssel der anfordernden Stelle übermitteln kann
die freigebende Stelle mit der anfordernden Stelle über ssl oder tls (ggf. als https) kommunizieren kann

37. Verfahren nach bisher genannten Ansprüchen, gekennzeichnet dadurch, daß die Kommunikation zwischen der freigebenden Stelle mit der anfordernden Stelle auf Netzwerk- und/oder Datenebene mit (unter anderem) einem oder mehreren der folgenden Schlüssel verschlüsselt und/oder authentifiziert werden kann:
Session-Key der kontextbezogenen Anforderung
Schlüsselpaar einer oder mehrerer der am Kontext der Anforderung beteiligten Personen
Schlüsselpaar des zur Anforderung verwendeten Systems
Schlüsselpaar des freigebenden Systems

38. Verfahren nach Anspruch 37, gekennzeichnet dadurch, daß anstatt einer direkten Verschlüsselung auch eines der in der Kryptographie bekannten Hybridverfahren verwandt werden kann

39. Verfahren nach bisher genannten Ansprüchen, gekennzeichnet dadurch, daß
freigebende und datenhaltende Stelle in einer Firewall-Topologie eingebaut sind
hierbei die datenhaltende Stelle im geschützten internen Bereich und die freigebende Stelle in der "DMZ" ("entmilitarisierten Zone") der Firewall liegen kann

40. Verfahren nach Anspruch 39, gekennzeichnet dadurch, daß die freigebende Stelle als "Application level Proxy" auf Schicht 7 des OSI-Referenzmodells realisiert ist

41. Verfahren nach bisher genannten Ansprüchen gekennzeichnet dadurch, daß
die Freigabe der Daten in mehreren Anforderungszyklen erfolgen kann
die Freigabe anfordernden Personen dabei weitere Kontextinformationen nachliefern können
die Freigabe erteilende Stelle Informationen über verfügbare Daten und/oder erforderliche weitere Kontextinformationen an die Freigabe anfordernde Stelle senden kann
durch wiederholte Zyklen der Umfang der freigegebenen Daten erweitert oder konkretisiert werden kann

42. Verfahren nach Anspruch 41 gekennzeichnet dadurch, daß
zur Vermittlung zwischen den Zyklen Hyperlinks, wie sie etwa aus dem WWW bekannt sind, verwendet werden können
diese Hyperlinks statisch festgelegt oder dynamisch von den beteiligten Systemen generiert werden können

43. Verfahren nach Anspruch 41 und 42, gekennzeichnet dadurch, daß die Hperlinks in Dokumenten folgender Struktur enthalten sein können:
Hypertext Markup Language (HTML)
Extended Markup Language (XML)

44. Verfahren nach Anspruch 41 bis 43, gekennzeichnet dadurch, daß zur Übertragung während der Anforderungszyklen eine Variante des Internet-Protokolls http (insbesondere auch https) verwendet wird

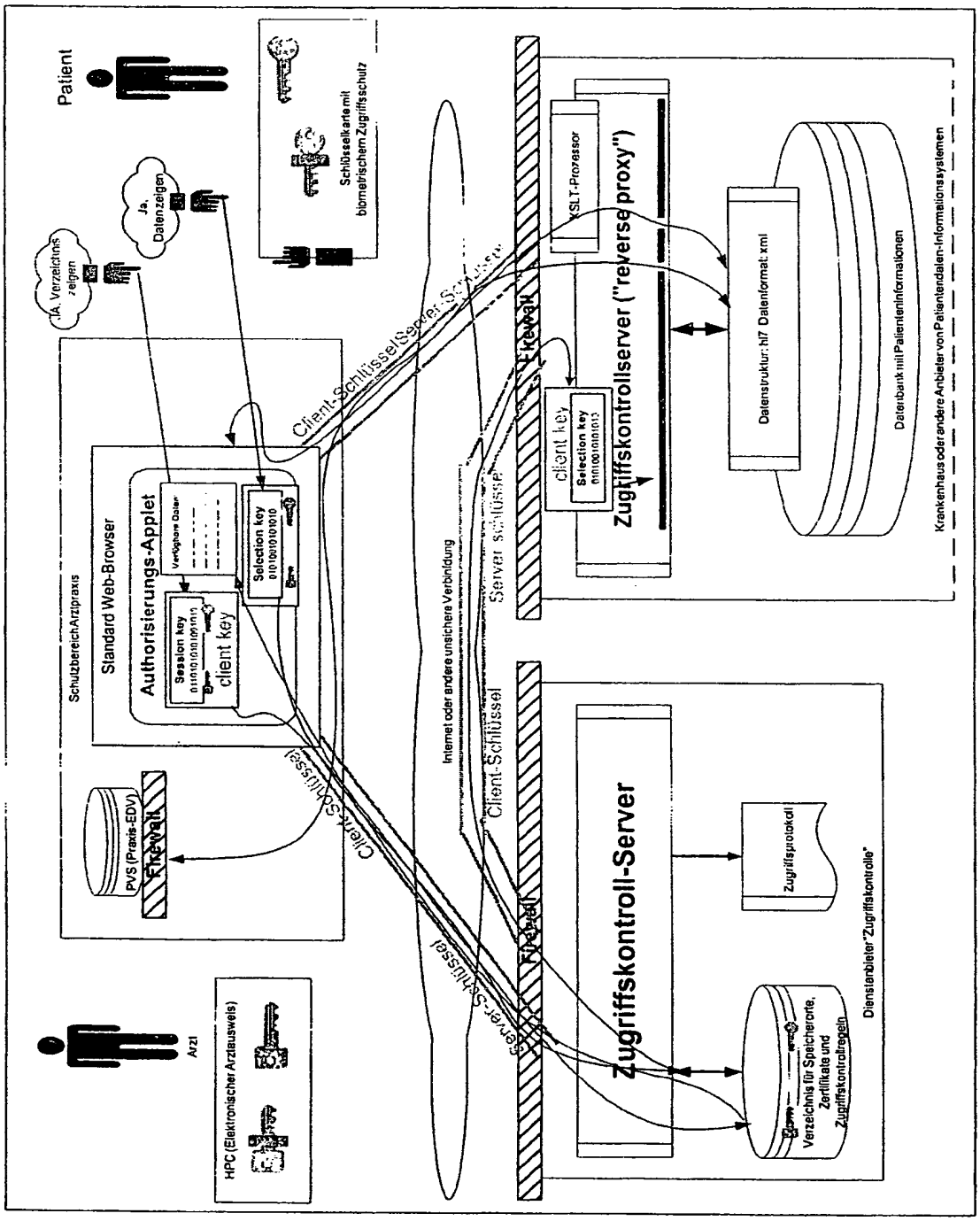
45. Verfahren nach bisher genannten Ansprüchen, gekennzeichnet dadurch, daß
an die Stelle von natürlichen Personen auch andere juristische Personen treten können
Personen auch andere vertreten können
die Rolle von Personen, die im eigenen Namen oder in Vertretung handeln, von datentechnischen Systemen übernommen werden kann, die innerhalb des beschriebenen Verfahrens das gleiche Verhalten zeigen, und deren Funktion davon abhängig ist, daß die vom System vertretene Person diese Handlung durch eine Willensäußerung dem System gegenüber kundgetan hat.

Hierzu 3 Seite(n) Zeichnungen

- Leerseite -

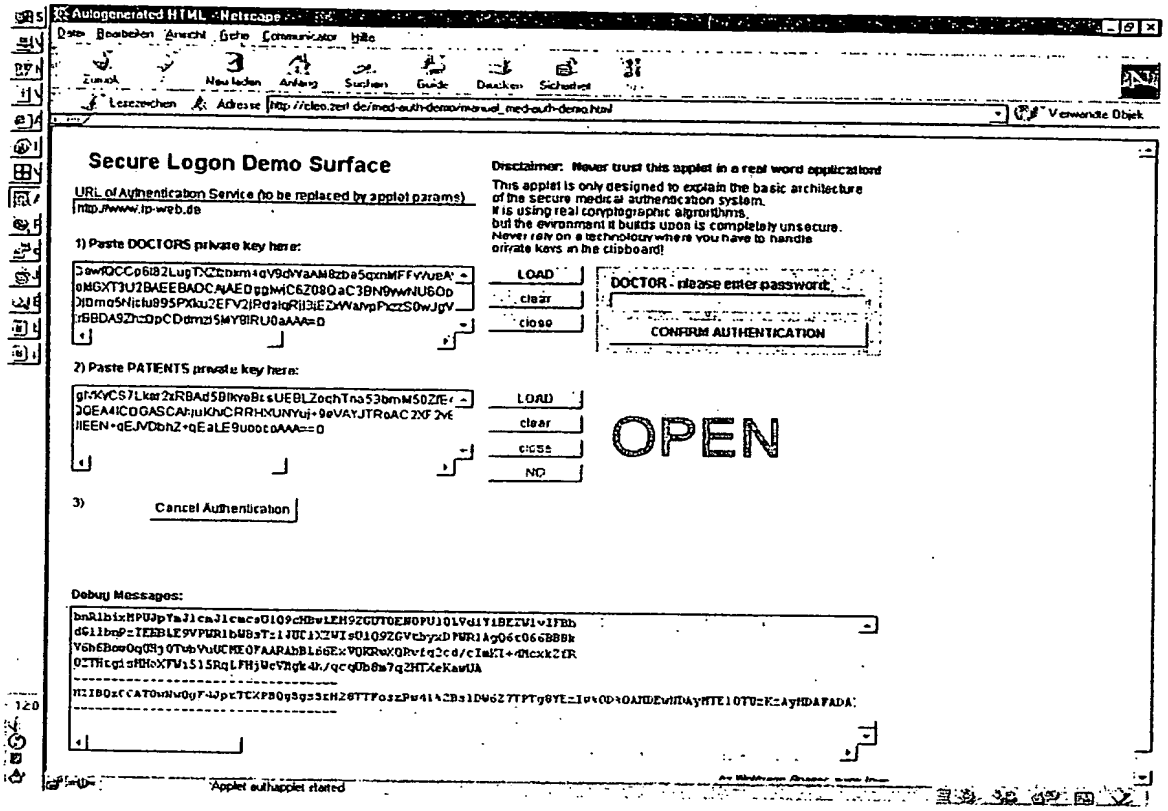
Nummer: DE 101 21 819 A1
Int. Cl.[7]: G 06 F 12/14
Offenlegungstag: 21. November 2002

Zeichnung 1 - Gesamtsystem:



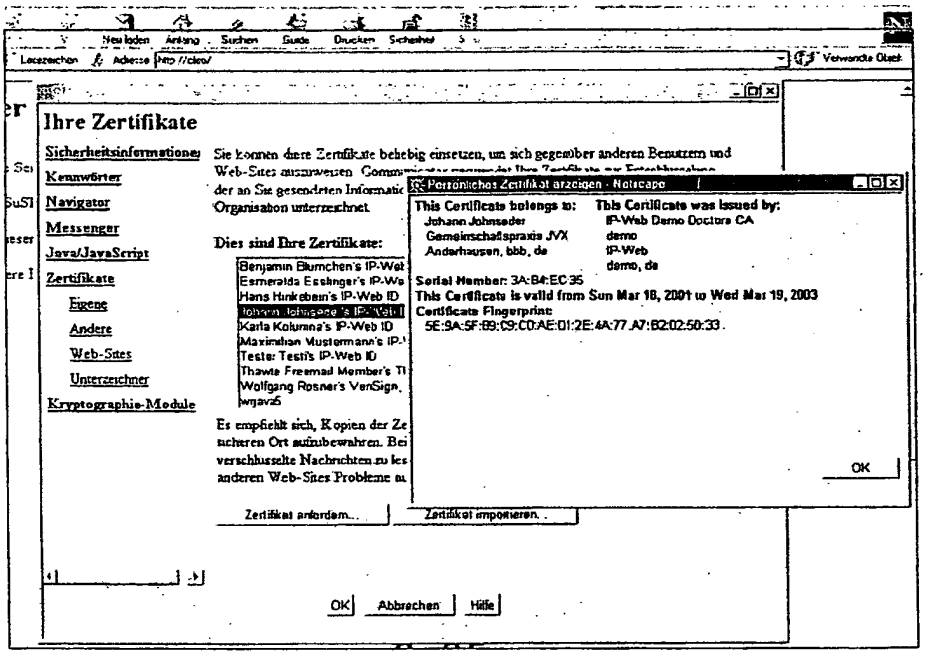
Nummer: DE 101 21 819 A1
Int. Cl.[7]: G 06 F 12/14
Offenlegungstag: 21. November 2002

Zeichnung 2 - Beispiel der Authentifizierungsoberfläche (Entwicklungsinstallation):



Nummer: DE 101 21 819 A1
Int. Cl.[7]: G 06 F 12/14
Offenlegungstag: 21. November 2002

Zeichnung 3 - Zertifikatsverwaltung in einem Standard-Browser:



This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☑ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.

As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.